Content Promotion for Online Content Platforms with Network Diffusion Effect

Yunduan Lin
Civil and Environmental Engineering Department, University of California, Berkeley, yunduan_lin@berkeley.edu

Mengxin Wang
Industrial Engineering and Operations Research Department, University of California, Berkeley, mengxin_wang@berkeley.edu

Zuo-Jun Max Shen
Industrial Engineering and Operations Research Department, University of California, Berkeley, maxshen@berkeley.edu

Heng Zhang
W. P. Carey School of Business, Arizona State University, hengzhang24@asu.edu

Renyu Zhang
NYU Shanghai, 1555 Century Avenue, Shanghai, China, 200122, renyu.zhang@nyu.edu

Online content platform aims to maximize content adoption which is simultaneously driven by the platform promotion and network diffusion. We study the candidate generation and promotion (CGP) problem for online content with the diffusion effect in this paper. Motivated by real-world datasets from the industry partner, we propose a novel diffusion model for online content that captures time variant diffusion effect and the existence of promoting probability. Based on the diffusion model, we formulate the CGP problem as a two-stage optimization problem. We show that the problem is NP-hard and can be seen as a submodular maximization problem. By utilizing the two-stage structure and combining the greedy idea from submodularity, we propose an $(1-\frac{1}{e})(1-\epsilon)$-approximation algorithm. We also consider the online version of this problem where the platform is simultaneously estimating the diffusion parameters and deploying the content promotion decisions. We propose a state-dependent non-anticipatory online policy that achieves $\tilde{\mathcal{O}}(\sqrt{T})$ of α-Cumulative Myopic Regret (α-CMRegret). Numerical results from a real-world dataset show that the diffusion process for online content deviates drastically from the traditional diffusion models and our proposed model provides much better fit. We also demonstrate the efficiency of our offline and online algorithms with the dataset.

Key words: Online content, Network diffusion effect, Submodular maximization, Online learning

1. Introduction

Online content platforms have met with considerable success in recent years. The proliferation of social media further facilitates the spread of online content. On online content platforms with social media, such as Facebook, YouTube, and Instagram, users create their own content and share within their social networks. For instance, YouTube which has more than two billion of monthly active users by 2020 and 500 hours of videos uploaded per minute utilizes a community tab to help users engage with their audience throughout the network. The recent Pew Research Center report

(Shearer and Mitchell 2021) indicates that 53% of U.S. adults get news from social media "often" or "sometimes". Among all social media sites, Facebook serves as a regular news source for 36% of U.S. adults and ranks first.

Online content includes various online information formats, such as online reviews, blogs and videos. A content is adopted by a user if the user clicks on it. The profit/reward of a online content platform is the total user adoptions. Online content platforms gain profits by providing users with suitable content. The platform tends to promote contents with a higher popularity historically, in the hope of attracting more future adoptions. Consequently, a large portion of the content with the potential to be popular via network diffusion is overlooked. This is called the *rich-get-richer* principle. As a result of the rich-get-richer principle, the majority of adoptions come from a small portion of contents. The time-sensitivity of online content promotion causes a more critical issue: if a content cannot attract enough adoptions within a short period of time, it will be replaced by newly uploaded contents and drop out of the promotion list of the platform. Therefore, the platform faces the following critical questions: What contents are more worthwhile to promote to get a higher total reward? How much effort should we spend on promoting each content? How to balance the promotion between content with high popularity and content with high adoption potential? These questions have drawn significant attention from both researchers and the industry. Recently, a range of models have been proposed to explain and characterize the popularity of content (Bakshy et al. 2011, Zhao et al. 2015, Rizoiu et al. 2017). However, the question of how to utilize such information to assist platform decision-making and drive revenue growth remains open.

Most past studies have focused solely on the *direct targeted effect* based on the user's preference or the *indirect diffusion effect* of a specific content. There are few prior works that address the revenue maximization problem while involving the network diffusion effect of all content. As indicated by Vahabi et al. (2015), the maximum number of items to be recommended is always limited at a time, and ignoring the underlying network diffusion effect may result in inefficient use of recommendation slots and harm the long-term revenue. Incorporating the network diffusion effect when promoting content is important. The diffusion models of new products have been widely studied in marketing literature and implemented in practice to predict the adoption level of a variety of consumption goods. However, these models are seldom used in the context of online platforms because the diffusion of online content exhibits a much more complicated structure.

In this study, we aim to fill this gap by developing a new diffusion-based promotion scheme for online content platforms. The scheme comprises two stages: candidate generation and promotion. In the candidate generation stage, a small subset of content is chosen from a large corpus. In the promotion stage, the platform allocates limited promotion opportunities to each candidate

content. The two-stage procedure windows down a large volume of content while maintaining a small portion of content that is attractive to the users. Because the reward of a content not only comes from its direct targeted effect, but also from the indirect diffusion effect, these two stages need to be treated as a whole to maximize the total adoption reward.

The diffusion process of online content largely depends on the promotion scheme of the platform. As a result, the diffusion curve of online content drastically deviates from traditional diffusion models for consumer goods. Motivated by this observation, we propose an online diffusion model (ODM) that captures the unique features of online content based on the widely adopted Bass diffusion model (BDM) (Bass 1969). We inherit the notion of *innovative* and *imitative* effects from BDM and introduce promoting probability, which controls the entire diffusion process, into the diffusion model. With the advent of information technology, we can keep track of all user behaviors and distinguish the users' roles as innovators or imitators. Therefore, we generalize the innovative and imitative effects from the population level to the individual level and incorporate diffusion heterogeneity over time. Real-world data experiments show that ODM well characterizes the adoption curve of online content.

Based on the diffusion model, we formulate the CGP problem as a two-stage optimization problem. We show that the offline CGP is NP-hard and develop an $(1-\frac{1}{e})(1-\epsilon)$-approximation algorithm that can solve the problem with time complexity $\mathcal{O}(\frac{|\mathcal{V}|}{\epsilon}\log(\frac{|\mathcal{V}|}{\epsilon})K)$. $\mathcal{V}$ is the set of video corpora and K denotes the size of the candidate set. ϵ is a hyperparameter that provides an operational balance between the execution time and approximation accuracy. The algorithm takes advantage of the greedy idea of submodular maximization and an efficient oracle to compute the marginal gain of each content. We then consider the online CGP problem and propose an adaptive policy that simultaneously learns the diffusion parameters and makes decisions. We define α-CMRegret for such an online learning problem and demonstrate that the α-CMRegret of our adaptive policy has an upper bound of $\tilde{O}(\sqrt{T})$.

Finally, we analyze and conduct experiments using data from a video-sharing platform.

1.1. Contribution and organization

Our contribution is summarized as follows:

- **A novel CGP modeling with network diffusion for online contents** A real dataset shows that the diffusion of online content is largely different from that of traditional consumer goods. Despite the considerable impact of online content, there are no existing models to characterize their diffusion processes. We propose the ODM to characterize the diffusion of online content. Specifically, ODM captures the heterogeneity of the adoption time as well as platform promotion intensity. To the best of our knowledge, our study is the first to propose such a diffusion model

and validate it using real-world data. We then model the CGP problem based on the proposed diffusion model. The CGP problem combines the candidate generation and promotion. Naturally, it can be decomposed into two stages: the first stage is a subset selection problem and the second stage is a continuous quadratic program.

- **Algorithmic design for CGP problem** We show that CGP problem is NP-hard and can be considered as a submodular maximization problem. The submodularity proof utilizes a two-stage structure by constructing a novel parameterized variant of the second-stage problem. We design offline and online algorithms for the CGP problem. We first provide an algorithm that optimally solves the parameterized second-stage problem within $\mathcal{O}(K \log K)$ time. Based on the submodularity, we integrate this algorithm with the greedy idea to provide an $(1-\frac{1}{e})(1-\epsilon)$ approximation algorithm for the first-stage problem that runs in $\mathcal{O}(\frac{|\mathcal{V}|}{\epsilon}\log(\frac{|\mathcal{V}|}{\epsilon})K)$. In the online setting when the diffusion parameters are not known to the platform, we propose an adaptive algorithm whose α-CMRegret is upper bounded by $\tilde{O}(\sqrt{T})$.
- **Numerical experiments with real datasets** We evaluate our models and algorithms using a real-world dataset from a large video-sharing platform and demonstrated the effectiveness of our method. First, we fit our proposed ODM with real content diffusion data. The ODM well-characterizes the real diffusion curve and has a smaller MAE than the traditional Bass-type model. Second, we use the dataset to evaluate the performance of both offline and online algorithms. In reality, the approximation ratio is usually not tight. Along the time horizon, our offline algorithm can achieve significant improvement over the benchmarks, and the online algorithm can learn the parameters in a short time period and performs much better than the benchmark.

The remainder of the paper is structured as follows: In Section 2, we review the related literature. In Section 3 and section 4, we propose the diffusion model and develop the CGP problem as well as offline algorithms. In Section 5, we investigate the problem in an online setting and analyze the performance of our proposed algorithm. Section 6 presents our numerical studies based on real-world data, followed by concluding statements in Section 7.

2. Literature Review

Our work draws on three branches of the literature, namely, diffusion models in social networks, online content promotion, and multi-armed bandits.

2.1. Diffusion models in social networks

Many diffusion models have been proposed in marketing and information system research, where aggregate-level and individual-level models complement each other (Rahmandad and Sterman 2008).

The aggregate-level model, pioneered by BDM (Bass 1969) and a large number of extensions (Easingwood et al. 1983, Horsky and Simon 1983, Norton and Bass 1987), captures the adoption trend with parsimonious differential equation. Theoretically, the aggregate-level model suffers from several shortcomings (Toubia et al. 2008). It does not explicitly provide the dynamics of how and why an individual adopts the diffusion process. In particular, its parameters always perform as scale coefficients but do not have a measurable definition (e.g. the probability of some events). On the contrary, individual-level models, consider the interaction among all entities in a specific network and can be as extreme as each user has idiosyncratic adoption behavior. There have been an increasing number of applications in recent decades, and we refer readers to review papers (Kiesling et al. 2012, Zhang and Vorobeychik 2019). One of the limitations of the individual-level diffusion model is that the diffusion process is not easily quantifiable through a simple formula; hence, the characterization of the market relies heavily on simulation techniques. Owing to such limitations, using individual-level models to forecast the total adoption after diffusion and further solving some optimization problems is impractical.

When considering diffusion in large-scale networks, especially in online social networks, overall adoption trends and individual choice behavior models are equally important to the platform. Recent research has attempted to combine the benefits of aggregate-level diffusion models and individual-level diffusion models. Rahmandad and Sterman (2008) and Macal (2010) developed the stochastic agent-based version of the classic epidemic models, which is a deterministic nonlinear differential equation model, and compared it with the example model having aggregate-level and individual-level counterparts. Goel et al. (2016) introduced a rigorous definition of structural virality and apply this measure to investigate the diffusion. In this paper, we provide a measurable definition of innovative and imitative parameters to assist with the identification of the diffusion mechanism.

2.2. Online content promotion

There is no clear definition of online content promotion, here, we refer to the external effort that an online content platform spends on increasing content adoption.

On one hand, recommender systems (RSs) are generally applied to gain a high direct targeted effect. Various recommendation algorithms (Resnick et al. 1994, Kitts et al. 2000, Covington et al. 2016) have been proposed to evaluate how users like a particular item among which the attributes of users, contents and activity are mainly considered. Recently, Azaria et al. (2013) and Lu et al. (2014) demonstrated that maximizing the immediate item relevancy does not align with utility maximization in RSs. Vahabi et al. (2015) was the first to mention that the availability of social network can empower the utility maximization of RSs. In other words, not only directly targeted

users contribute to the total adoption, but also the users that are influenced by them contributes to it. They proposed a social-diffusion-aware RS that can efficiently use recommendation slots and enhance the overall performance.

On the other hand, the stream of influence maximizing (IM) literature considers the indirect diffusion effect. Kempe et al. (2003) first modeled IM as an algorithmic problem and proposed the seeding strategy under the Independent Cascade Model and Linear Threshold Model. Arora et al. (2017) and Li et al. (2018) provided a comprehensive experimental study and survey about IM, respectively. However, IM only considers the case wherein the platform aims to popularize a particular content and is unable to deal with the promotion of all contents. In our setting, we consider candidate generation and promotion jointly for all contents and propose a two-stage optimization model to maximize total adoption across the network.

2.3. Multi-armed bandits

Our work in online optimization is also related to multi-armed bandits (MAB), that is widely used to balance the exploration-exploitation trade-off when the estimation and decision need to be made simultaneously.

Several studies (Li et al. 2010, Song et al. 2014, Zeng et al. 2016) have applied MAB framework to an online recommender system. ϵ-greedy, upper confidence bound (UCB), and Thompson sampling algorithms are typically used in these applications. In our setup, the CGP problem can be viewed as a combinatorial bandit problem (CMAB) (Chen et al. 2013). However, our problem is different from the classical bandit setting with an i.i.d. assumption in the following aspects: (i) different arms may share the same pair of parameters and the number of arms increases with time; (ii) the reward of an arm or a super arm that comprises a set of arms depends on the history actions and changes over time; and (iii) the mean of reward does not possess a typical distribution or a specific dynamics such as the Markov decision process (MDP). Thus, we generalize the CMAB problem with a cumulative myopic regret setting to shed light on the online problem.

3. Online Diffusion Model

In this section, we formally introduce the ODM. We start by providing a motivating observation from a real-world dataset and then describe the ODM in detail.

3.1. Preliminaries of BDM

The discrete time counterpart of the BDM can be written as

$$a(t) = \left(p + \frac{q}{m} A(t-1)\right)(m - A(t-1))$$

where $a(t)$ denotes the number of new adopters at t, $A(t-1) = \sum_{\tau=1}^{t-1} a(\tau)$ is the number of cumulative adopters through time period $t-1$; p and q are the innovative coefficient and imitative

coefficient, respectively; and m is the market share. In BDM, users can be classified as innovators or imitators. At each time step, the number of users facing the adoption decision depends on the remaining market potential, and the fraction of new innovators remains the same while the fraction of new imitators is proportional to the potential influential sources, which denotes the number of accumulated adopters.

3.2. Motivation from a real-world dataset

The idea of innovativeness in previous research often refers to the degree to which an individual or other unit of adoption is relatively earlier in adopting new ideas than other members of a system (Rogers 2010). For online content, we can explicitly define the innovators and imitators according to whether the adoption happens after exposure from platform. The diffusion of a content is naturally driven by two forces: (i) Innovative effect: the platform can directly expose the content to users and users will choose whether to adopt it according to their intrinsic value; (ii) Imitative effect: after some users adopt the content, they may share it with their friends or inspire other users to search for the content. Hence other users may become exposed to the content without being targeted directly and will have the chance to adopt it. The data from a large video sharing platform show that around 27.73% of adoptions come from imitative effect.

However, when using BDM to fit the diffusion process, the diffusion curve deviates drastically from the true diffusion process as shown in Figure 1. Traditional Bass-type models for consumer goods cannot fully capture the diffusion behavior of online contents. Basically, previous diffusion models ignore the impact of platform behavior on users adoption. This motivates us to develop a diffusion model tailored for online content. In the remainder of this section, we first provide some preliminary knowledge of the BDM for completeness. Subsequently, we introduce our proposed ODM based on the notion of BDM. In Section 6, we show that the ODM successfully characterizes the true diffusion curve of online content using a real dataset.

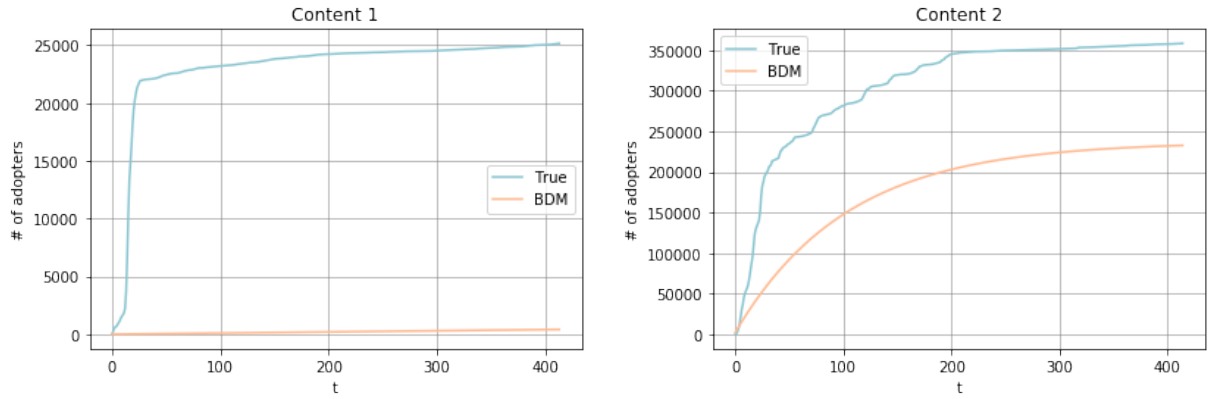

Figure 1 Diffusion curves for two online contents and the corresponding fitted BDM curves

3.3. Diffusion Model

We consider the diffusion process unfolding in discrete steps and diffusion occurs after direct exposure at each time step.

We assume that the market size m is equal to the total number of users on the platform and does not change over time. We use p and q to denote the intensity of the innovative and imitative effects on all users in the network. To be precise, the innovative adoption of a user follows the Bernoulli distribution with success probability p when the content is directly exposed to the user. Imitative adoption follows Bernoulli distribution with success probability $\frac{q}{m}\left(\tilde{A}(t-1)+a^{\text{in}}(t)\right)$ which is proportional to the sum of the exponentially smoothed number of previous adopters $\tilde{A}(t-1)=\sum_{\tau=1}^{t-1}\gamma^{t-\tau}a(\tau)$ (γ denotes the discount factor with $0\leq\gamma\leq 1$) and the current new innovators $a^{\text{in}}(t)$. When $\gamma=1$, the imitative effect is proportional to the exact number of adopters in the market, which is the same case in BDM; when $\gamma=0$, only the innovators incur diffusion during at the same time period. Both p and q are assumed to be between zero and one.

The traditional diffusion model such as BDM implicitly assume that each non-adopter confronts with the adoption decisions at every time step. However, as the platform has the power to trigger innovative adoption, it can also control the diffusion process by varying the promotion intensity. We adopt a fluid modeling approach widely used in the analysis of stochastic systems. In particular, we use the promoting probability $x(t)$ to represent the platform strategy. Consider that at time t, the platform will expose the content to the user with probability $x(t)$. To avoid user fatigue by repetitive exposures, we enforce the probability to be no larger than the proportion of non-adoptors within the type when at most one content can be received by users at each time t, that is, $x(t)\leq 1-\frac{A(t-1)}{m}$. We generalize the case when more than one content can be received within one time period in later sections. Under the assumptions of our model, adoption decisions for being an innovator are independently made by every user watching the content in each period, while the decisions for being an imitator depend on adoption history. Let $\mathcal{H}(t)=\{(a^{\text{in}}(\tau),a^{\text{im}}(\tau))\}_{\tau=1}^{t-1}$ be the observed adoption history up to time t. To calculate the conditional expected number of new adopters in t, we first derive the conditional expected number of new innovators and imitators given the adoption history and promoting probability as (1) and (2).

$$\mathbb{E}[a^{\text{in}}(t)|\mathcal{H}(t-1),x(t)]=\mathbb{E}[a^{\text{in}}(t)|A(t-1),x(t)]=mpx(t) \tag{1}$$

$$\begin{aligned}\mathbb{E}\left[a^{\text{im}}(t)|\mathcal{H}(t-1),x(t)\right]&=\mathbb{E}\left[\frac{q}{m}\left(\tilde{A}(t-1)+a^{\text{in}}(t)\right)\left(m-A(t-1)-a^{\text{in}}(t)\right)\middle|\mathcal{H}(t-1),x(t)\right]\\&=\frac{q}{m}\left[(\tilde{A}(t-1)+mpx(t))(m-A(t-1)-mpx(t))-mp(1-p)x(t)\right]\end{aligned} \tag{2}$$

Basically, the ODM differs from the BDM in three aspects: (i) Stochasticity: BDM assumes a deterministic process at an aggregate level that cannot capture the uncertainty when users make

their own decisions. We extend the diffusion model to a stochastic setting that can explicitly model each user's behavior. (ii) Non-uniformity in time: The assumption that the imitative effect is proportional to the number of adopters is no longer suitable for online content. This assumption implies that all adopters have uniform weights of diffusion incentive; however, as often is the case, users are more likely to diffuse the content shortly after their adoption and the incentive will diminish as time goes by. To better characterize this property, discounted weights should be assigned to adopters according to their different adoption times. (iii) Limited innovative exposure: As numerous contents compete for the limited exposure slots, the number of users facing adoption decisions not only depends on the remaining market potential but also largely depends on the promotion probability which is determined by the platform. It is necessary to model the platform behavior during the diffusion process.

4. Problem Setup

In the previous section, we describe the diffusion model for a single online content. The platform can show the content to as many users as possible to maximize the adoption of the specific content. However, numerous contents in the corpus competing for exposure opportunities in reality and how to decide the promoting probability of each content becomes a non-trivial problem. In this section, we formulate the static optimization problem to determine and promote the candidate contents to achieve the optimal total reward from all the contents.

4.1. Static optimization: offline CGP problem

In this section, we consider the static optimization problem at each time step t. When the static optimization is considered, the time argument t is ignored in the notation. The granularity of time steps in our problem is approximately hourly or daily. In other words, the update of the candidate set is not that frequent because there is no need to change the decision in reality when the adoption level only changes slightly. Thus, the offline CGP problem should be considered based on the current adoption status of all contents in the existing content set $\mathcal{V}$. Only the contents included in the candidate set can be exposed to users via innovative effects. Once a user adopts a certain content, they drop out of the potential adopter population and enter the adopter population, thereby influencing others' adoption of the same content through imitative effect.

The platform makes a two-stage decision. At the beginning of each time period, the platform selects a subset V of at most K contents from $\mathcal{V}$ as the candidate set; during this period, the platform distributes contents included in the candidate set to users and each user can receive L contents within the period. L is also called as slot restriction. We define two sets of decision variables for this two-stage decision: the selected candidate set V, and the set of continuous variables x_v representing the promotion probability of content v for each promotion slot.

Hereafter, we use the bold notation to denote the collection of a particular variable for all contents in the vector form. The formulation of the optimization problem depends on the current diffusion state of all the contents which is represented by $(\mathbf{A}, \tilde{\mathbf{A}})$. When the diffusion state is $(\mathbf{A}, \tilde{\mathbf{A}})$, given promotion probability $\boldsymbol{x}$, we define the expected reward as the expected adoption from both new innovators and imitators. Following the calculation in (1) and (2), the expected total reward can be derived as

$$\mathbb{E}\left[\sum_{v\in\mathcal{V}} a_v^{\text{in}} + a_v^{\text{im}} \middle| \boldsymbol{A}, \tilde{\boldsymbol{A}}, \boldsymbol{x}\right] = \sum_{v\in\mathcal{V}} \left(-\lambda_v x_v^2 + \mu_v x_v\right) + \sum_{v\in\mathcal{V}} q_v \left(1 - \frac{A_v}{m}\right) \tilde{A}_v$$

where $\lambda_v = mL^2 q_v p_v^2$ and $\mu_v = Lq_v p_v[\frac{m}{q_v} + m - (1-p_v) - (A_v + \tilde{A}_v)]$, respectively. Although the values of $\boldsymbol{\lambda}, \boldsymbol{\mu}$ depend on the diffusion state $(\boldsymbol{A}, \tilde{\boldsymbol{A}})$, these arguments can be considered as fixed and given when dealing with static optimization. Given $\boldsymbol{A}, \tilde{\boldsymbol{A}}$, $\sum_{v\in\mathcal{V}} q_v(1 - \frac{A_v}{m})\tilde{A}_v$ is a constant that does not affect the optimization. Define

$$R(\boldsymbol{x}) = \sum_{v\in\mathcal{V}} -\lambda_v x_v^2 + \mu_v x_v, \tag{3}$$

we can write the offline CGP problem as follows:

$$\begin{aligned}
\underset{\boldsymbol{x}, V}{\text{maximize}} \quad & R(\boldsymbol{x}) && \text{(4a)}\\
\text{subject to} \quad & |V| \le K, && \text{(4b)}\\
& \sum_{v\in\mathcal{V}} x_v = 1, && \text{(4c)}\\
& 0 \le x_v \le \frac{1}{L}\left(1 - \frac{A_v}{m}\right) \cdot \mathbb{1}\{v \in V\}, \forall v \in \mathcal{V}. && \text{(4d)}
\end{aligned}$$

(4b) is the capacity constraint, that ensures at most K contents can be selected. (4c) is the probability constraint, which requires the sum of promoting probability equals 1. (4d) is a set of novelty constraints that provides the lower and upper bound of the promoting probability. The content should have a non-negative promoting probability and can only be positive if it is included in the candidate set. Moreover, to reduce repeated recommendations of the same content, we restrict the total expected numbers of promotions to be not larger than the number of non-adopters.

The offline CGP problem involves both discrete and continuous decision variables and a nonlinear objective function. This problem is highly complicated and there are no obvious optimization algorithms besides enumeration to solve it. This becomes an issue especially when the content set $\mathcal{V}$ is large, which is common in real practice. In fact, the offline CGP problem is NP-hard in general as shown in the following. This result motivated us to look for an efficient approximation algorithm that can handle large amounts of content.

First, we show the monotonicity of objective function $R(\boldsymbol{x})$ in Lemma 1. Lemma 1 facilitates the proof of NP-hardness as well as the submodularity in the next section.

LEMMA 1. *For any diffusion state* $(\boldsymbol{A}, \tilde{\boldsymbol{A}})$*,* $R(\boldsymbol{x})$ *is increasing in each* x_v *in its feasible region.*

Proof of Lemma 1: When $A_v = m$ which means all the users have adopted content v, x_v can only take the value 0. When $A_v < m$, we have

$$\begin{aligned}
\frac{\partial R(\boldsymbol{x})}{\partial x_v} &= -2L^2 m q_v p_v^2 x_v + L q_v p_v \left[\frac{m}{q_v} + m - (1-p_v) - (A_v + \tilde{A}_v)\right] \\
&\geq L q_v p_v \left[-2p_v(m - A_v) + \frac{m}{q_v} + m - (1-p_v) - (A_v + \tilde{A}_v)\right] && \text{(5a)} \\
&\geq L q_v p_v \left[2(1-p_v)(m - A_v) - (1-p_v) - m + \frac{m}{q_v}\right] && \text{(5b)} \\
&\geq L q_v p_v m \left(\frac{1}{q_v} - 1\right) && \text{(5c)} \\
&\geq 0 && \text{(5d)}
\end{aligned}$$

(5a) follows from the feasible region defined in constraints (4d). (5b) follows from $\tilde{A}_v \leq \gamma A_v < A_v$. Since A_v is an integer, $A_v < m$ can also be represented as $m - A_v \geq 1$. (5c) follows since innovative effect p_v is between 0 and 1 and (5d) follows since imitative effect q_v is between 0 and 1. □

Our proof of NP-hardness then utilizes Lemma 1 to construct a reduction from the SUBSET-SUM problem, which is known to be NP-hard. The proof is presented in Appendix A.

THEOREM 1. *The offline CGP problem is NP-hard.*

4.2. Parameterized variants of offline CGP problem

In this section, we analyze the second-stage problem of determining the probability of promotion when a candidate set V is given. We show the structural properties of second-stage optimal solutions. By exploiting the structural properties, we develop an $\mathcal{O}(|V| \log |V|)$ algorithm that efficiently determines the second-stage promotion decision. Moreover, these properties play a critical role in deriving the submodularity of the first-stage problem in Section 4.3.

We construct a collection of parameterized optimization problems when the coefficients of the novelty constraints and the candidate set are given as parameters for the original problem (4) and propose an efficient algorithm to optimally solve a single parameterized instance. Let $\overline{\boldsymbol{s}}$ be the right-hand side coefficients of constraint (4d) in the original problem, that is, $\overline{s}_v = \frac{1}{L}(1 - \frac{A_v}{m})$ for all $v \in \mathcal{V}$. We define the parameters of the parameterized optimization problem as a tuple $(\boldsymbol{s}, V)$ which comprises a vector $\boldsymbol{s} \in \mathcal{S}$ and a subset $V \subseteq \mathcal{V}$, where $\mathcal{S} = \{\boldsymbol{s} \in \mathbb{R}^{|\mathcal{V}|} : 0 \leq s_v \leq \overline{s}_v, \forall v \in \mathcal{V}\}$. We refer to the optimization instance with parameter $(\boldsymbol{s}, V)$ as $\mathcal{I}(\boldsymbol{s}, V)$, and let $\boldsymbol{x}^*(\boldsymbol{s}, V)$ and $U(\boldsymbol{s}, V)$ be the optimal solution and value, respectively. Therefore, $\mathcal{I}(\boldsymbol{s}, V)$ can be explicitly formulated as

$$\begin{aligned}
\underset{\boldsymbol{x}}{\text{maximize}} \quad & \sum_{v \in \mathcal{V}} -\lambda_v x_v^2 + \mu_v x_v \\
\text{subject to} \quad & \sum_{v \in \mathcal{V}} x_v = 1, \\
& 0 \leq x_v \leq s_v \cdot \mathbb{1}\{v \in V\}, \forall v \in \mathcal{V}
\end{aligned}$$

For an arbitrary instance $\mathcal{I}(\boldsymbol{s}, V)$, we define the marginal reward of content $v \in \mathcal{V}$ as the partial derivative of the objective function $R(\boldsymbol{x})$ at the optimal solution $x_v^*(\boldsymbol{s}, V)$. Mathematically, let $\eta_v(\boldsymbol{s}, V)$ denote the marginal reward of content $v \in \mathcal{V}$, and $\eta_v(\boldsymbol{s}, V) = -2\lambda_v x_v^*(\boldsymbol{s}, V) + \mu_v$. A direct observation is that if the optimal solution $x_v^*(\boldsymbol{s}, V)$ increases, the marginal reward $\eta_v(\boldsymbol{s}, V)$ will decrease. Furthermore, the generalized marginal reward of $\mathcal{I}(\boldsymbol{s}, V)$ is defined as

$$\tilde{\eta}(\boldsymbol{s}, V) = \inf_{v: x_v^*(\boldsymbol{s}, V) > 0} \eta_v(\boldsymbol{s}, V) \tag{6}$$

When $\sum_{v \in \mathcal{V}} s_v \cdot \mathbb{1}(v \in V) = 0$, set $\{v : x_v^*(\boldsymbol{s}, V) > 0\}$ is empty and the infimum of an empty set is $+\infty$ by default. $\tilde{\eta}(\boldsymbol{s}, V)$ can characterize the optimal solution of instance $\mathcal{I}(\boldsymbol{s}, V)$ to some extent and we will provide the property of $\tilde{\eta}(\boldsymbol{s}, V)$ in Lemma 2.

LEMMA 2. *For any instance $\mathcal{I}(\boldsymbol{s}, V)$ such that $\tilde{\eta}(\boldsymbol{s}, V) < +\infty$, the following properties hold for all $v \in V$:*

(i) If $x_v^*(\boldsymbol{s}, V) > 0$, $\eta_v(\boldsymbol{s}, V) \geq \tilde{\eta}(\boldsymbol{s}, V)$.

(ii) If $x_v^*(\boldsymbol{s}, V) < s_v$, $\eta_v(\boldsymbol{s}, V) \leq \tilde{\eta}(\boldsymbol{s}, V)$.

(iii) For all content v such that $0 < x_v^*(\boldsymbol{s}, V) < s_v$, their marginal rewards are the same, and $\eta_v(\boldsymbol{s}, V) = \tilde{\eta}(\boldsymbol{s}, V)$.

Proof of Lemma 2: We will show the proof for a fixed instance $\mathcal{I}(\boldsymbol{s}, V)$ in the following and thus ignore the parameters $(\boldsymbol{s}, V)$ in notation. Proof of (i) is directly shown by definition.

Proof of (ii) and (iii): We will proof by contradiction. Without loss of generality, let v_i be the content such that $\tilde{\eta} = \eta_{v_i} = -2\lambda_{v_i} x_{v_i}^* + \mu_{v_i}$. Assume there exists a content v_j such that $x_{v_j}^* < s_{v_j}$ and $\eta_{v_j} > \tilde{\eta}$. Consider a solution $\boldsymbol{x}$ that is given by

$$x_v = \begin{cases} x_v^*, & \text{when } v \notin \{v_i, v_j\} \\ x_{v_i}^* - \epsilon, & \text{when } v = v_i \\ x_{v_j}^* + \epsilon, & \text{when } v = v_j \end{cases}$$

where ϵ is a positive number. Since $x_{v_i}^* > 0$ by definition and $x_{v_j}^* < s_{v_j}$, there exists $\epsilon > 0$ that makes $\boldsymbol{x}$ a feasible solution to $\mathcal{I}(\boldsymbol{s}, V)$. Consequently, for a feasible $\boldsymbol{x}$, we have

$$\begin{aligned} R(\boldsymbol{x}) - R(\boldsymbol{x}^*) &= \sum_{v \in V} -\lambda_v x_v^2 + \mu_v x_v - \left(\sum_{v \in V} -\lambda_v x_v^{*2} + \mu_v x_v^* \right) \\ &= \sum_{v \in \{v_i, v_j\}} -\lambda_v x_v^2 + \mu_v x_v - \left(\sum_{v \in \{v_i, v_j\}} -\lambda_v x_v^{*2} + \mu_v x_v^* \right) \\ &= (\eta_{v_j} - \eta_{v_i})\epsilon - \left(\lambda_{v_i} + \lambda_{v_j}\right) \epsilon^2 \end{aligned}$$

Since $\eta_{v_i} < \eta_{v_j}$, there exists $\epsilon > 0$ that makes $R(\boldsymbol{x}) - R(\boldsymbol{x}^*) > 0$ which contradicts with the optimality of $\boldsymbol{x}^*$. Thus, for any $x_{v_j}^* < s_{v_j}$, we have $\eta_{v_j} \leq \tilde{\eta}$. Property (ii) holds.

When $0 < x_{v_j}^* < s_{v_j}$, we additionally have $\eta_{v_j} \geq \tilde{\eta}$ by definition. Property (iii) holds. $\square$

Based on the property of $\tilde{\eta}(\boldsymbol{s}, V)$ in Lemma 2, we can have the following Proposition 1 and Proposition 2 for the optimal value of $\mathcal{I}(\boldsymbol{s}, V)$. The detailed proofs are shown in Appendix A.

PROPOSITION 1. *For any instance $\mathcal{I}(\boldsymbol{s}, V)$, there exists a unique optimal solution $\boldsymbol{x}^*(\boldsymbol{s}, V)$.*

On one hand, Lemma 2 shows the relation between $\eta_v(\boldsymbol{s}, V)$ and $\tilde{\eta}(\boldsymbol{s}, V)$ based on $x_v^*(\boldsymbol{s}, V)$. On the other hand, once the value of $\tilde{\eta}(\boldsymbol{s}, V)$ is known, the optimal solution $\boldsymbol{x}^*(\boldsymbol{s}, V)$ can be determined directly. For a specific instance $\mathcal{I}(\boldsymbol{s}, V)$, as $x_v^*(\boldsymbol{s}, V)$ can only take value in $[0, s_v]$, we define the lower bound and upper bound of $\eta_v(\boldsymbol{s}, V)$ to be $\underline{\eta}_v(\boldsymbol{s}, V) = -2\lambda_v s_v + \mu_v$ and $\overline{\eta}_v(\boldsymbol{s}, V) = \mu_v$ correspondingly. Hence, suppose the value of $\tilde{\eta}(\boldsymbol{s}, V)$ is known beforehand, all the contents $v \in V$ can be categorized into 3 sets, that is, $\underline{V}(\boldsymbol{s}, V) = \{v \in V : \eta_v(\boldsymbol{s}, V) < \tilde{\eta}(\boldsymbol{s}, V)\}$, $\tilde{V}(\boldsymbol{s}, V) = \{v \in V : \eta_v(\boldsymbol{s}, V) = \tilde{\eta}(\boldsymbol{s}, V)\}$ and $\overline{V}(\boldsymbol{s}, V) = \{v \in V : \eta_v(\boldsymbol{s}, V) > \tilde{\eta}(\boldsymbol{s}, V)\}$. Meanwhile, the optimal solution $x_v^*(\boldsymbol{s}, V)$ can be explicitly derived as

$$x_v^*(\boldsymbol{s}, V) = \begin{cases} 0 & \text{when } v \in \underline{V}(\boldsymbol{s}, V) \\ \frac{\mu_v - \tilde{\eta}(\boldsymbol{s}, V)}{2\lambda_v} & \text{when } v \in \tilde{V}(\boldsymbol{s}, V) \\ s_v & \text{when } v \in \overline{V}(\boldsymbol{s}, V) \end{cases} \tag{7}$$

Apart from knowing $\tilde{\eta}(\boldsymbol{s}, V)$ can recover the optimal solution directly, we can notice that three sets $\underline{V}(\boldsymbol{s}, V)$, $\tilde{V}(\boldsymbol{s}, V)$, $\overline{V}(\boldsymbol{s}, V)$ and a solution $\boldsymbol{x}(\boldsymbol{s}, V)$ can also be defined for an arbitrary η in the same way. We use the superscript η to denote the corresponding sets and variables.

PROPOSITION 2. *For any instance $\mathcal{I}(\boldsymbol{s}, V)$ that satisfies $\sum_{v \in V} s_v > 1$, equality* (8) *holds if and only if $\eta = \tilde{\eta}(\boldsymbol{s}, V)$, which is the generalized marginal reward.*

$$\sum_{v \in \overline{V}^\eta} s_v + \sum_{v \in \tilde{V}^\eta} \frac{\mu_v - \eta}{2\lambda_v} = 1 \tag{8}$$

Propositions 1 and 2 show the uniqueness of $\tilde{\eta}(\boldsymbol{s}, V)$ and they imply that as long as we can find a η that satisfies (8), we can derive the optimal solution. Therefore, we propose Algorithm 1 to obtain the optimal solution of $\mathcal{I}(\boldsymbol{s}, V)$, reducing the polynomial time complexity of a general convex quadratic programming to $\mathcal{O}(|V| \log |V|)$.

For the case when $\sum_{v \in V} s_v \leq 1$, we directly have the result that $\boldsymbol{x}^*(\boldsymbol{s}, V) = \boldsymbol{s}$. In other cases, we have $\sum_{v \in V} x_v^*(\boldsymbol{s}, V) = 1$ holds and $\tilde{\eta}(\boldsymbol{s}, V)$ can only take values in $[\underline{\eta}(\boldsymbol{s}, V), \overline{\eta}(\boldsymbol{s}, V)]$, where $\underline{\eta}(\boldsymbol{s}, V) = \max_{v \in V} \underline{\eta}_v(\boldsymbol{s}, V)$ and $\overline{\eta}(\boldsymbol{s}, V) = \max_{v \in V} \overline{\eta}_v(\boldsymbol{s}, V)$. As we can divide the interval $[\underline{\eta}(\boldsymbol{s}, V), \overline{\eta}(\boldsymbol{s}, V)]$ into a number of subintervals such that the sets $\underline{V}(\boldsymbol{s}, V)$, $\tilde{V}(\boldsymbol{s}, V)$ and $\overline{V}(\boldsymbol{s}, V)$ do not change as long as $\tilde{\eta}$ stays within that subinterval. Thus, we only need to enumerate the subintervals and find the one that contains the valid $\tilde{\eta}(\boldsymbol{s}, V)$. We can construct $(2|V| - 1)$ subintervals of $[\underline{\eta}(\boldsymbol{s}, V), \overline{\eta}(\boldsymbol{s}, V)]$ using the thresholds in $\{\overline{\eta}_v(\boldsymbol{s}, V) : v \in V\} \cup \{\underline{\eta}_v(\boldsymbol{s}, V) : v \in V\}$. If we enumerate the subintervals in order which requires $\mathcal{O}(|V| \log |V|)$ for sorting, whether the valid $\tilde{\eta}(\boldsymbol{s}, V)$ is contained in each subinterval can be checked within $\mathcal{O}(1)$ time since only one element will change among three sets.

Algorithm 1 Find the optimal solution for $\mathcal{I}(\boldsymbol{s}, V)$

Input: Parameter $\boldsymbol{s}$, V.

Result: Optimal solution $\boldsymbol{x}^*$ and optimal value U.

if $\sum_{v\in V} s_v \le 1$ **then**
  $\boldsymbol{x}^* := \boldsymbol{s}$, $U := R(\boldsymbol{x}^*)$.
else
  Sort $\{\underline{\eta}_v : v \in V\} \cup \{\overline{\eta}_v : v \in V\}$ in non-decreasing order as $\eta' = \{\eta_{[1]}, \eta_{[2]}, \ldots, \eta_{[2|V|]}\}$.
  Initialize $\overline{V} := V$, $\tilde{V} := \emptyset$, $\underline{V} := \emptyset$.
  Define $\beta_1 := \sum_{v\in\tilde{V}} \frac{\mu_v}{2\lambda_v}$, $\beta_2 := \sum_{v\in\tilde{V}} \frac{1}{2\lambda_v}$, $\beta_3 := \sum_{v\in\overline{V}} s_v$.
  for $i = 1 \to 2|V| - 1$ **do**
    if $\eta_{[i]} \in \{\underline{\eta}_v : v \in V\}$ **then**
      Without loss of generality, assume $\eta_{[i]} = \underline{\eta}_w$.
      w drops out of $\overline{V}$ and enters $\tilde{V}$. $\overline{V} := \overline{V} - \{w\}$, $\tilde{V} := \tilde{V} + \{w\}$.
      Update parameters $\beta_1 := \beta_1 + \frac{\mu_w}{2\lambda_w}$, $\beta_2 := \beta_2 + \frac{1}{2\lambda_w}$, $\beta_3 := \beta_3 - s_w$.
    else
      Without loss of generality, assume $\eta_{[i]} = \overline{\eta}_w$.
      w drops out of $\tilde{V}$ and enters $\underline{V}$. $\tilde{V} := \tilde{V} - \{w\}$, $\underline{V} := \underline{V} + \{w\}$.
      Update parameters $\beta_1 := \beta_1 - \frac{\mu_w}{2\lambda_w}$, $\beta_2 := \beta_2 - \frac{1}{2\lambda_w}$.
    end
    $\tilde{\eta} := \frac{\beta_1+\beta_3-1}{\beta_2}$.
    if $\eta_{[i]} \le \tilde{\eta} \le \eta_{[i+1]}$ **then**
      Derive $\boldsymbol{x}^*$ as (7), $U := R(\boldsymbol{x}^*)$. **break**.
    end
  end
end
return $\boldsymbol{x}^*, U$.

The optimality of the solution found by Algorithm 1 directly follows from the Proposition 1 and Proposition 2.

4.3. Submodularity and algorithm

In this section, we show that the offline CGP problem is submodular with respect to the selected content using parameterized variants. Based on the submodularity, we propose an algorithm based on the greedy idea and improve the algorithm by indicating some approaches to reduce the execution time.

4.3.1. Proof of submodularity To show the submodularity, we need to analyze the relation between different parameterized instances. We first establish Theorem 2 to show that the param-

eterized collection of optimization problems has a monotone optimal solution for nested subset instances.

THEOREM 2. *For all $\boldsymbol{s} \in \mathcal{S}$, and $V_1 \subseteq V_2 \subseteq \mathcal{V}$, $x_v^*(\boldsymbol{s}, V_1) \geq x_v^*(\boldsymbol{s}, V_2)$ holds for all $v \in V_1$.*

Proof of Theorem 2: We will show the proof for a pair of fixed instances $\mathcal{I}(\boldsymbol{s}, V_1)$ and $\mathcal{I}(\boldsymbol{s}, V_2)$ in the following and thus ignore the parameter $\boldsymbol{s}$ in notation.

Let V' be the set of contents $v \in V_1$ that has different optimal solutions in $\mathcal{I}(\boldsymbol{s}, V_1)$ and $\mathcal{I}(\boldsymbol{s}, V_2)$, that is, $V' = \{v \in V_1 : x_v^*(V_1) \neq x_v^*(V_2)\}$. We only need to consider the case when $V' \neq \emptyset$ in the following proof, and $\tilde{\eta}(V_1) < \infty$ and $\tilde{\eta}(V_2) < \infty$ always hold in this case.

Consider a solution $\boldsymbol{x}$ that is given by

$$x_v = \begin{cases} x_v^*(V_2), & \text{when } v \in V_1 \\ 0, & \text{when } v \notin V_1 \end{cases}$$

It is obvious that $\boldsymbol{x}$ is a feasible solution of $\mathcal{I}(V_1)$. By Lemma 1, we know that there exists $w \in V'$ such that $x_w^*(V_1) > x_w = x_w^*(V_2)$. $x_w^*(V_1) > x_w^*(V_2)$ directly implies that $\eta_w(V_1) < \eta_w(V_2)$, $x_w^*(V_1) > 0$ and $x_w^*(V_2) < s_w$. By Lemma 2, we have

$$\tilde{\eta}(V_1) \leq \eta_w(V_1) < \eta_w(V_2) \leq \tilde{\eta}(V_2) \tag{9}$$

For all $v \in V' \backslash \{w\}$, if $x_v^*(V_1) = s_v$ or $x_v^*(V_2) = 0$, it is trivial that $x_v^*(V_2) < x_v^*(V_1)$. Otherwise, if $x_v^*(V_1) < s_v$ and $x_v^*(V_2) > 0$, we have

$$\eta_v(V_1) \leq \tilde{\eta}(V_1) \text{ and } \tilde{\eta}(V_2) \leq \eta_v(V_2)$$

Both inequalities follow from Lemma 2. We can thus have $\eta_v(V_1) < \eta_v(V_2)$ following (9), which further implies $x_v^*(V_1) > x_v^*(V_2)$.

In conclusion, $x_v^*(V_1) \geq x_v^*(V_2)$ for all $v \in V_1$ when $V_1 \subseteq V_2 \subseteq \mathcal{V}$. $\square$

COROLLARY 1. *For all $\boldsymbol{s} \in \mathcal{S}$, and $V_1 \subseteq V_2 \subseteq \mathcal{V}$, $\tilde{\eta}(\boldsymbol{s}, V_1) \leq \tilde{\eta}(\boldsymbol{s}, V_2)$ holds if $\sum_{v \in V_1} s_v \geq 1$.*

The Corollary 1 follows immediately from Theorem 2. When $\sum_{v \in V_1} s_v \geq 1$, if there exists $w \in V_2 \backslash V_1$ such that $x_w^*(\boldsymbol{s}, V_2) > 0$, at least one of $w \in V_1$ satisfies $x_w^*(\boldsymbol{s}, V_1) \neq x_w^*(\boldsymbol{s}, V_2)$. Consequently, inequality (9) also holds. Otherwise, if for all $v \in V_2 \backslash V_1$, $x_v^*(\boldsymbol{s}, V_2) = 0$ holds, we can conclude that $\boldsymbol{x}^*(\boldsymbol{s}, V_1) = \boldsymbol{x}^*(\boldsymbol{s}, V_2)$ which leads to $\tilde{\eta}(\boldsymbol{s}, V_1) = \tilde{\eta}(\boldsymbol{s}, V_2)$.

Based on the characterization of two instances with nested candidate sets, we then show that $U(\boldsymbol{s}, V)$ is submodular with regard to set V. Our original offline CGP problem is a special case when the coefficients are equal to $\overline{\boldsymbol{s}}$.

THEOREM 3. *$U(\boldsymbol{s}, V)$ is a monotone submodular set function with regard to V for any $\boldsymbol{s} \in \mathcal{S}$.*

Proof of Theorem 3: The complete proof is provided in Appendix A. We provide the proof outline as follows.

It is trivial that $U(\boldsymbol{s}, V_1) \leq U(\boldsymbol{s}, V_2)$ when $V_1 \subseteq V_2$, thus, $U(\boldsymbol{s}, V)$ is monotone.

To show the submodularity, it suffices to show that

$$U(\boldsymbol{s}, V_2 + \{w\}) - U(\boldsymbol{s}, V_2) \leq U(\boldsymbol{s}, V_1 + \{w\}) - U(\boldsymbol{s}, V_1)$$

where $V_1 \subseteq V_2$ and $w \in \mathcal{V} \backslash V_2$. Since adding a content w to the selected subset is equivalent to lifting s_w from 0 to s_w, we just need to show that the increasing reward of lifting one entry of parameter $\boldsymbol{s}$ is diminishing for nested subsets. Let $\boldsymbol{e}_v$ denote the unit vector where the entry that represents content v is 1 and 0 otherwise, we just need to show the following inequality holds.

$$U(\boldsymbol{s}, V_2 + \{w\}) - U(\boldsymbol{s} - s_w \boldsymbol{e}_w, V_2) \leq U(\boldsymbol{s}, V_1 + \{w\}) - U(\boldsymbol{s} - s_w \boldsymbol{e}_w, V_1) \tag{10}$$

We can decompose the left hand side (LHS) and right hand side (RHS) of (10) into three parts by the value of parameter for content w. The two critical values for partition are $a_1 = 1 - \sum_{v \in V_1} s_v$ and $a_2 = x_w^*(\boldsymbol{s} + (\overline{s}_w - s_w)\mathbf{e}_w, V_2 + \{w\})$. Without loss of generality, we consider the case when $0 < a_1 < a_2 < s_w$, and then show the inequality holds for each part separately. Other cases can be easily generalized. We denote three inequalities in a simplified notation as

(i) $\Delta U(\boldsymbol{s}, 0, a_1, V_2, w) \leq \Delta U(\boldsymbol{s}, 0, a_1, V_1, w)$.

(ii) $\Delta U(\boldsymbol{s}, a_1, a_2, V_2, w) \leq \Delta U(\boldsymbol{s}, a_1, a_2, V_1, w)$.

(iii) $\Delta U(\boldsymbol{s}, a_2, s_w, V_2, w) \leq \Delta U(\boldsymbol{s}, a_2, s_w, V_1, w)$.

(i) and (iii) can be proved by combining Theorem 2 and some basic properties of convex optimization. The proof of (ii) applies Corollary 1 to show that for every slight perturbation on the parameter, the inequality holds.

By summing up (i), (ii), (iii), we can show the submodularity of $U(\boldsymbol{s}, V)$. □

We now conclude that the offline CGP problem (4) is equivalent to a monotone submodular maximization problem with a cardinality constraint. The well-known greedy algorithm (Nemhauser et al. 1978) provides an $(1 - \frac{1}{e})$-approximation. The idea of the greedy algorithm is to iterate K times and choose the content having the largest marginal gain at each iteration. The oracle complexity of the algorithm is $\mathcal{O}(|\mathcal{V}|K)$. When Algorithm 1 is applied for the oracle computation, the total time complexity becomes $\mathcal{O}(|\mathcal{V}|K^2 \log K)$ because every time the oracle is queried, the selected set has a cardinality not larger than K.

4.3.2. Improvement with the greedy idea Benefiting from the ordered structure of Algorithm 1, the total runtime complexity can be further reduced by combining the greedy idea. Because the greedy algorithm adds one more content into the chosen candidate set at each time, we can

utilize the information from previous iterations to speed up the calculation. In particular, let the selected set at iteration k be V_k. The list of partition values is known as $\eta'(\boldsymbol{s}, V_k)$ and the generalized marginal reward is $\tilde{\eta}(\boldsymbol{s}, V_k)$. We show the Iterative Algorithm 1 for an instance $\mathcal{I}(\boldsymbol{s}, V_k + \{w\})$ where $w \in \mathcal{V} \backslash V_k$ at iteration $k+1$.

Iterative Algorithm 1 at Iteration k

(i) Instead of sorting in Line 4 of Algorithm 1, perform binary search in $\eta'(\boldsymbol{s}, V_k)$ and insert $\underline{\eta}_w = -2\lambda_w s_w + \mu_w$ and $\overline{\eta}_w = \mu_w$ to get $\eta'(\boldsymbol{s}, V_k + \{w\})$.

(ii) Instead of enumerating over all $2k+1$ subintervals in Line 7 of Algorithm 1, only consider the subintervals that has the value not less than $\tilde{\eta}(\boldsymbol{s}, V_k)$.

As the majority of $\eta'(\boldsymbol{s}, V_k + \{w\})$ remains the same, the binary search in (i) reduces the runtime complexity of sorting from $\mathcal{O}(K \log K)$ to $\mathcal{O}(\log K)$ and thus no longer dominates the entire procedure. (ii) reduces the search space for $\tilde{\eta}(\boldsymbol{s}, V_k + \{w\})$ because by Corollary 1 adding a new content into the candidate set will not decrease $\tilde{\eta}$. In other words, only searching in a space that is greater than or equal to $\tilde{\eta}(\boldsymbol{s}, V_k)$ can also guarantee optimality. Ideally, the number of subintervals will be far less than $2k+1$, but the complexity remains $\mathcal{O}(K)$. Overall, the time complexity of Iterative Algorithm 1 is $\mathcal{O}(|\mathcal{V}|K^2)$.

However, the pure greedy algorithm may still not work well when both the number of contents and the capacity increase. In the following, we combine the decreasing threshold algorithm in Badanidiyuru and Vondrák (2014) and CELF in Leskovec et al. (2007) to propose a Diminishing-Threshold and Lazy-Update greedy (DTLU) algorithms in Algorithm 2 that achieves an approximation ratio of $(1-\frac{1}{e})(1-\epsilon)$. Here, ϵ is a hyperparameter that can be set by the platform according to the accuracy requirement. This algorithm is supposed to reduce the execution time by decreasing the number of times the value query is performed while maintaining a good approximation performance.

Diminishing-Threshold aims to obtain the balance between runtime efficiency and approximation rate. Instead of selecting the element with the largest marginal gain, we now include all elements that can gain an additional reward greater than a threshold d at each iteration and discount the threshold to $(1-\epsilon)d$ in the next iteration. Badanidiyuru and Vondrák (2014) has shown that such algorithm can achieve an $(1-\frac{1}{e})(1-\epsilon)$-approximation solution with $\mathcal{O}(\frac{n}{\epsilon}\log\frac{n}{\epsilon})$ oracle complexity.

Lazy-Update further reduces the number of queries. Consider the selected set V_k at iteration k and V_{k+1} at iteration $k+1$, for any $v \notin V_{k+1}$, we have

$$U(V_{k+1} + \{v\}) - U(V_{k+1}) \leq U(V_k + \{v\}) - U(V_k)$$

Depending on the submodularity, the marginal gain of a content in the current iteration cannot be better than that in the previous iterations. Thus, there is no need to update the marginal gain for

every content at each iteration. By recording the marginal gain of contents from the first iteration, we only need to recalculate the value when it exceeds the current threshold. In our setting, the lazy update is quite useful. We consider the case wherein a content has been adopted by a large portion of users; its marginal gain in the first iteration will perform as a threshold itself and avoid extra computation. This part has no impact on the approximation ratio.

Algorithm 2 DTLU greedy algorithm

Input: set of content $\mathcal{V}$, capacity K and approximation parameter ϵ.

Result: $\boldsymbol{x}^*$ promoting probability.

for $v \in \mathcal{V}$ **do**
  Initialize the marginal gain. $g_v := U(\overline{\boldsymbol{s}}, \{v\})$.
end
$V := \emptyset$, $U^* := 0$.
Initialize the threshold to be the largest marginal gain. $g_{\max} := \max_{v\in\mathcal{V}} g_v$, $d := g_{\max}$.
while $d \geq \frac{\epsilon}{|\mathcal{V}|} g_{\max}$ **do**
  for $v \in \mathcal{V} \backslash V$ **do**
    if $|V| < K$ *and* $g_v \geq d$ **then**
      Calculate $U(V + \{v\})$ by Iterative Algorithm 1, and update $g_v := U(V + \{v\}) - U^*$.
      if $g_v \geq d$ **then**
        Include v into candidate set. $V := V + \{v\}$.
      end
    end
  end
  $\gamma := \gamma(1 - \epsilon)$.
end
Calculate $\boldsymbol{x}^*(V)$ by Iterative Algorithm 1.
return $\boldsymbol{x}^*(V)$.

5. Online Learning

In the previous section, we assumed that the innovative and imitative effect parameters were known and studied the offline problem. In this section, we develop an adaptive policy that simultaneously learns diffusion parameters from user feedback and optimizes candidate set selection and promotion.

In traditional market, distinguishing between innovators and imitators is difficult, and the only observable data is the total adoption in a certain time period. Hence, the parameters p and q are always estimated by regression analysis as a whole. However, with the advent of information technology, it is not as demanding as previously to trace user footprints in the digital market. For an online platform, it is easy to capture the exposure from the system as well as the sharing between users and further identify whether an adoption is motivated by other users. Thus, we

can estimate p and q separately from the abundant historical data which makes the online CGP problem possible.

5.1. Online CGP problem

With social media, users as well as the platform continuously generate new content that enriches the content corpus and makes the online problem much more challenging.

We consider the problem in a discrete time horizon $\mathcal{T} := [1, \dots, T]$. At each time period $t \in \mathcal{T}$, the content corpus can be represented as $\mathcal{V}_t$. In reality, users generate content at a steady frequency. This is characterized by Assumption 1:

ASSUMPTION 1. *The size of content set $|\mathcal{V}_t|$ grows linearly with time t.*

To facilitate the learning process, we assume there is a family of categories Ω such that each content belongs to one of the categories $\omega \in \Omega$, and let $S_\omega(t)$ denote the set of contents that belong to category $\omega \in \Omega$ at time t. Thus, for each category $\omega \in \Omega$, we have a set of innovative parameters p_ω and an imitative parameter q_ω.

At the beginning of each period, the platform updates the innovative and imitative parameters estimated from historical user feedback, selects the candidate of contents and obtains the optimal policy to distribute the content to different users. Let $N_v(t)$ and $P_v(t)$ denote the total number of times users have been directly exposed to content v and adopted via platform promotion by time t respectively. Let $M_v(t)$ and $Q_v(t)$ denote the number of non-adopters before imitation occurs and imitators of content v by time t, respectively. We further define the potential scale of imitation as

$$\tilde{M}_v(t) = \left[\sum_{\tau=1}^{t-1} \gamma^{t-\tau} \left(M_v(\tau - 1) - M_v(\tau) \right) + P_v(t) - P_v(t-1) \right) \right] M_v(t)$$

By summing up the corresponding variables of all contents in a specific category, we can extend the notion of $N, P, M, \tilde{M}, Q$ to a category $\omega \in \Omega$ as follows: $N_\omega(t) = \sum_{v \in S_\omega(t)} N_v(t)$, $P_\omega(t) = \sum_{v \in S_\omega(t)} P_v(t)$, $M_\omega(t) = \sum_{v \in S_\omega(t)} M_v(t)$, $\tilde{M}_\omega(t) = \sum_{v \in S_\omega(t)} \tilde{M}_v(t)$, and $Q_\omega(t) = \sum_{v \in S_\omega(t)} Q_v(t)$.

The unbiased estimators for diffusion parameters are shown in Lemma 3.

LEMMA 3. *$\hat{p}_\omega(t) = \frac{P_\omega(t)}{N_\omega(t)}$ is an unbiased estimator for p_ω and $\hat{q}_\omega(t) = \frac{Q_\omega(t) m}{\sum_{\tau=1}^{t} \tilde{M}_\omega(\tau)}$ is an unbiased estimator for q_ω.*

Using Lemma 3, we define optimistic estimators as follows using the UCB.

$$p_\omega^{\mathrm{OP}}(t) = \min\{\hat{p}_\omega(t) + \sqrt{\frac{2 \log t}{N_\omega(t)}}, 1\} \tag{11a}$$

$$q_\omega^{\mathrm{OP}}(t) = \min\{\hat{q}_\omega(t) + \frac{m \sqrt{2 \log t \sum_{\tau=1}^{t} M_\omega(\tau)}}{\sum_{\tau=1}^{t} \tilde{M}_\omega(\tau)}, 1\} \tag{11b}$$

As the diffusion of can only occur when there are existing adopters, we make Assumption 2 to ensure that the diffusion starts for each category after the initialization step.

ASSUMPTION 2. *After initialization step, at least one of the contents in each category $\omega \in \Omega$ has the number of adopters between 2 and $m-2$. Note that $2 \le m-2$ implies $m \ge 4$. Without loss of generality, for the initialization step, $\tilde{\boldsymbol{A}} = \boldsymbol{A}$.*

We then propose Adaptive-DTLU algorithm for online CGP problem as shown in Algorithm 3.

Algorithm 3 `Adaptive-DTLU algorithm for online CGP problem`

Input: set of initial content $\mathcal{V}_0$, number of slots L

Initialize $p_\omega^{\mathrm{OP}}(t) := 1$ and $q_\omega^{\mathrm{OP}}(t) := 1$ for all $\omega \in \Omega$, $t = 1$.
Initialize the promotion by promoting contents from each category $\omega \in \Omega$.
while $t < T$ **do**
  Update the content corpus $\mathcal{V}_t$.
  Compute $\boldsymbol{x}^*$ with parameters $p_\omega^{\mathrm{OP}}(t)$ and $q_\omega^{\mathrm{OP}}(t)$ according to Algorithm 2.
  for $\ell := 1 \to L$ **do**
    Platform exposes content v to each user with probability x_v^*.
  end
  for *Any feedback from the promotion* **do**
    Update $p_\omega^{\mathrm{OP}}(t)$ according to Equation (11a).
  end
  for *Any feedback from sources other than promotion* **do**
    Update $q_\omega^{\mathrm{OP}}(t)$ according to Equation (11b).
  end
end

5.2. Regret analysis of Adaptive-DTLU algorithm

In this section, we analyze the performance of the Adaptive-DTLU Algorithm. Regret is widely used for the performance analysis of online learning policy, and is usually defined as the expected reward difference between the optimal decision and the exact choice.

However, the conventional regret is not appropriate for our problem for three reasons: (i) Because the offline problem is NP-hard and an approximation oracle is used to solve the problem, comparing the outcome of the online algorithm with the exact optimal value is no longer fair; (ii) our problem has time-variant rewards for the same actions that depend on the current adoption state and only a state-dependent non-anticipatory policy can be given even in the offline case; (iii) no further assumptions are made for newly generated contents which makes it intractable to obtain the global optimal along the time horizon. Inspired by the α-approximation regret (Chen et al. 2013) in CMAB with an α-approximation oracle and cumulative myopic regret (Ahmed et al. 2017) for the MDP,

we define an α-cumulative myopic regret (α-CMRegret) for the state-dependent non-anticipatory policy as follows

$$\alpha\text{-CMRegret}^{\pi}(T) = \mathbb{E}_{\pi}\left[\sum_{t=1}^{T} \alpha R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}^*; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}_t; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t)\right] \tag{12}$$

where the function $R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t)$ is the online generalization of (3) when the diffusion state is $(\boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t)$ and the innovative and imitative coefficients are $\boldsymbol{p}$ and $\boldsymbol{q}$, respectively. $\boldsymbol{x}_t$ is the solution provided by our online policy at time t and $\boldsymbol{x}^*$ is the optimal solution when the true parameters $\boldsymbol{p}$ and $\boldsymbol{q}$ are known to the platform. α-CMRegret is a variant of regret that denotes the cumulative maximal possible improvements if the true parameters are known to the platform.

To analyze α-CMRegret, we first establish the following lemmas to bound the reward deviations critical for regret analysis. The proofs are included in Appendix B.

Lemma 4 shows the concentration bound on $p_{\omega}^{\mathrm{OP}}(t)$ and $q_{\omega}^{\mathrm{OP}}(t)$.

LEMMA 4. *For any $N_\omega(t)$, $M_\omega(t)$ and $\tilde{M}_\omega(t)$, we have*

$$\mathbb{P}\left(p_{\omega}^{OP}(t) - 2\sqrt{\frac{2\log t}{N_\omega(t)}} \le p_\omega(t) \le p_{\omega}^{OP}(t)\right) \ge 1 - \frac{2}{t^4}$$

$$\mathbb{P}\left(q_{\omega}^{OP}(t) - \frac{2m}{\sum_{\tau=1}^{t}\tilde{M}_\omega(\tau)}\sqrt{\frac{2\log t}{\sum_{\tau=1}^{t} M_\omega(\tau)}} \le q_\omega(t) \le q_{\omega}^{OP}(t)\right) \ge 1 - \frac{2}{t^4}$$

When comparing two vectors, let $\ge$ (respectively, $\le$) be element-wise greater (respectively, less) than or equal to. Lemma 5 shows that the difference of optimal values of CGP problems with different diffusion parameters can be upper bounded by a linear function of the difference between parameters.

LEMMA 5. *There exists a linear function $f(\Delta\boldsymbol{p}, \Delta\boldsymbol{q}; \boldsymbol{A}, \tilde{\boldsymbol{A}})$, such that for any given pair of $\boldsymbol{p}'$, $\boldsymbol{q}'$, when given diffusion state $(\boldsymbol{A}, \tilde{\boldsymbol{A}})$, we have $R_{\boldsymbol{p}',\boldsymbol{q}'}(\boldsymbol{x}; \boldsymbol{A}, \tilde{\boldsymbol{A}}) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}; \boldsymbol{A}, \tilde{\boldsymbol{A}}) \le f(\Delta\boldsymbol{p}, \Delta\boldsymbol{q}; \boldsymbol{x})$ if $\Delta\boldsymbol{p} = \boldsymbol{p}' - \boldsymbol{p} \ge \boldsymbol{0}$ and $\Delta\boldsymbol{q} = \boldsymbol{q}' - \boldsymbol{q} \ge \boldsymbol{0}$ for all $\boldsymbol{x}$ where $\boldsymbol{p}$ and $\boldsymbol{q}$ are true diffusion parameters.*

Lemma 6 shows that under Assumption 2, the regret of imitative adoptions can be bounded by that of innovative adoptions.

LEMMA 6. *For $t \in \mathcal{T}$, we have*

$$\sqrt{\frac{1}{N_\omega(t) - N_\omega(1)}} \ge \frac{m}{\sum_{\tau=1}^{t-1}\tilde{M}_\omega(\tau)}\sqrt{\frac{1}{\sum_{\tau=1}^{t-1} M_\omega(\tau)}}$$

In Theorem 4, we characterize the regret bound of the Adaptive-DTLU algorithm.

THEOREM 4. *The α-CMRegret of Adaptive-DTLU policy during time T is bounded by*

$$\alpha\text{-}CMRegret(T) = O(\sqrt{mLT\log T})$$

Proof of Theorem 4: The complete proof is provided in Appendix B. We provide the proof outline as follows.

1. Define the "large" probability event of each parameter as $E_\omega(t) = \{p_\omega^{\mathrm{OP}}(t) - 2\sqrt{\frac{2\log t}{N_\omega(t)}} \le p_\omega(t) \le p_\omega^{\mathrm{OP}}(t)\}$ and $F_\omega(t) = \{q_\omega^{\mathrm{OP}}(t) - 2m\frac{\sqrt{2\log t\sum_{\tau=1}^t M_\omega(\tau)}}{\sum_{\tau=1}^t M_\omega(\tau)} \le q_\omega(t) \le q_\omega^{\mathrm{OP}}(t)\}$. Let $\zeta(t) = \bigcap_{\omega\in\Omega} E_\omega(t) \bigcap_{\omega\in\Omega} F_\omega(t)$ denoting the clean event when the "large" probability events hold simultaneously at time t.

2. Consider the clean event. Using the optimality guarantee of the offline policy, the single period regret can be upper bounded by

$$
\begin{aligned}
&\alpha R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}^*; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}_t; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t)\\
=&\alpha R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}^*; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p}^{\mathrm{OP}}(t),\boldsymbol{q}^{\mathrm{OP}}(t)}(\boldsymbol{x}_t; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t) + R_{\boldsymbol{p}^{\mathrm{OP}}(t),\boldsymbol{q}^{\mathrm{OP}}(t)}(\boldsymbol{x}_t; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}_t; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t)\\
\le&\alpha R_{\boldsymbol{p}^{\mathrm{OP}}(t),\boldsymbol{q}^{\mathrm{OP}}(t)}(\boldsymbol{x}^*; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t) - \alpha R_{\boldsymbol{p}^{\mathrm{OP}}(t),\boldsymbol{q}^{\mathrm{OP}}(t)}(\boldsymbol{x}_t^*; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t) + R_{\boldsymbol{p}^{\mathrm{OP}}(t),\boldsymbol{q}^{\mathrm{OP}}(t)}(\boldsymbol{x}_t; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}_t; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t)\\
\le&R_{\boldsymbol{p}^{\mathrm{OP}}(t),\boldsymbol{q}^{\mathrm{OP}}(t)}(\boldsymbol{x}_t; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}_t; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t)
\end{aligned}
$$

By Lemma 5 and Lemma 6, we can have the conditional regret of clean event to be

$$
\begin{aligned}
&\mathbb{E}_\pi\left[\sum_{t=1}^T \alpha R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}^*; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}_t; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t)\Big|\bigcap_{t=1}^T \zeta(t)\right]\\
&\le \mathbb{E}_\pi\left[\sum_{t=1}^T f(\boldsymbol{p}^{\mathrm{OP}}(t) - \boldsymbol{p}, \boldsymbol{q}^{\mathrm{OP}}(t) - \boldsymbol{q}; \boldsymbol{x}_t)\Big|\bigcap_{t=1}^T \zeta(t)\right]\\
&\le C\sqrt{\log T}\mathbb{E}_\pi\left[\sum_{\omega\in\Omega}\sum_{t=1}^{T-1} mL\Big(\sqrt{\frac{1}{N_\omega(t)}} + \sqrt{\frac{1}{N_\omega(t+1) - N_\omega(1)}}\Big)\right]
\end{aligned}
$$

where C is a constant. By the fact that the total number of times that contents of category ω is promoted to users at time t is less than or equal to mL and $\sum_{\omega\in\Omega} N_\omega(T) \le mLT$, we can further bound it by $\mathcal{O}(\sqrt{mLT\log T})$.

3. Consider the "bad" event. Using the union bound to calculate the probability of "bad" event, we have

$$
\mathbb{P}\left((\bigcap_{t=1}^T \zeta(t))^{\complement}\right) \le \sum_{t=1}^T\sum_{\omega\in\Omega} \mathbb{P}\left(E_\omega(t)^{\complement}\right) + \mathbb{P}\left(F_\omega(t)^{\complement}\right) \le C'\frac{|\Omega|}{T^3}
$$

Combine with Assumption 1, we can have the expected reward bounded by $\mathcal{O}(\frac{m|\Omega|}{T})$.

In conclusion,

$$
\alpha\text{-CMRegret} = \mathcal{O}(\sqrt{mLT\log T})
$$

□

As we consider a centralized problem for the entire market, α-CMRegret measures the total regret of all users. In a traditional MAB setting, a single user arrives at each time and thus the regret only refers to a single user. To make the upper bound of our regret more comparable to the traditional case, we can average over the market size and it becomes

$$\alpha\text{-CMRegret}^{\text{ave}} = \mathcal{O}(\sqrt{\frac{LT\log T}{m}}) \tag{13}$$

The conclusion here also shows that the upper bound becomes smaller when the market size increases.

6. Numerical Results

6.1. Data overview

For the experiment, we used a dataset from a video-sharing platform in China. The dataset comprise of the behavior log data of 227,986 short videos from over 174 million users for 20 days (7/1/2020-7/20/2020). The raw log data includes the timestamped records of videos exposed to users and the responses in terms of clicks. We can distinguish the adoption source of whether the user is exposed to the video via a platform or other resources.

In addition, the dataset contained information on the video category. In our experiment, we selected the videos that were newly uploaded to the platform during the time from 301 categories, indicating that the records contain the complete diffusion process for these contents. The content category encapsulates what the video is about, i.e. travel, education, and game.

6.2. Model calibration

In this section, we show the distribution of the time decay hyperparameters and diffusion parameters of real-world online content. We consider the time granularity of data samples to be hourly, i.e., each time period is an hour.

Hyperparameter γ: After collecting the number of innovators and imitators at each time period for different contents, we tuned the hyperparameter γ and predicted the corresponding number of imitators for each data sample. By calculating the mean absolute error (MAE) between the true value and predicted value, we select the value of γ that has the smallest MAE. In our case, $\gamma = 0.75$ as shown in Figure 2.

Diffusion parameters p **and** q: Given $\gamma = 0.75$, we derive the unbiased estimator for innovative coefficient p and imitative coefficient q according to Lemma 3. Heterogeneity was observed among the content categories.

The distributions of p and q are shown in Figure 3. From the left subfigure, no obvious positive correlation ($\rho = 0.106$) is observed between p and q for the same content category, implying that the content that attracts users directly via promotion may not necessarily lead to a higher diffusion

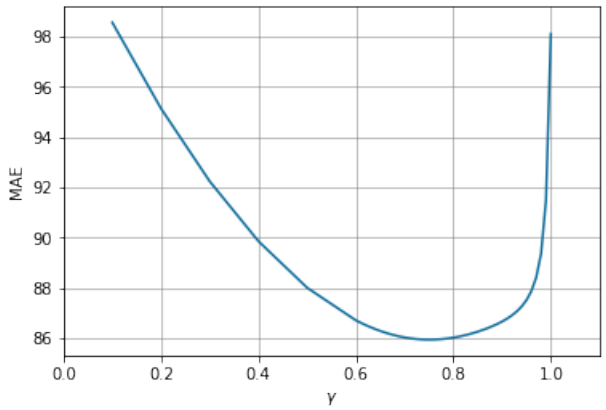

Figure 2 **MAE of number of imitators versus γ**

effect. From the right subfigure, we focus on the relative values of two parameters. As in traditional BDM for consumption goods, p is usually far less than q, with an average value of $p = 0.03$ and $q = 0.38$ (Sultan et al. 1990). In our model for online content, p has a wider range and higher value than q, with the average value of $p = 0.23$ and $q = 0.07$. However, the result does not mean that the imitative effect for online content is smaller than that for consumption goods. The low cost of adopting online content encourages users to click the contents in a timely manner when exposed to the platform, while the rapid update of online content makes it difficult to become consistently attractive. Thus, innovators account for a much larger proportion of adopters compared to consumption goods. Both scenarios emphasize the important role of promotion decisions in the ODM.

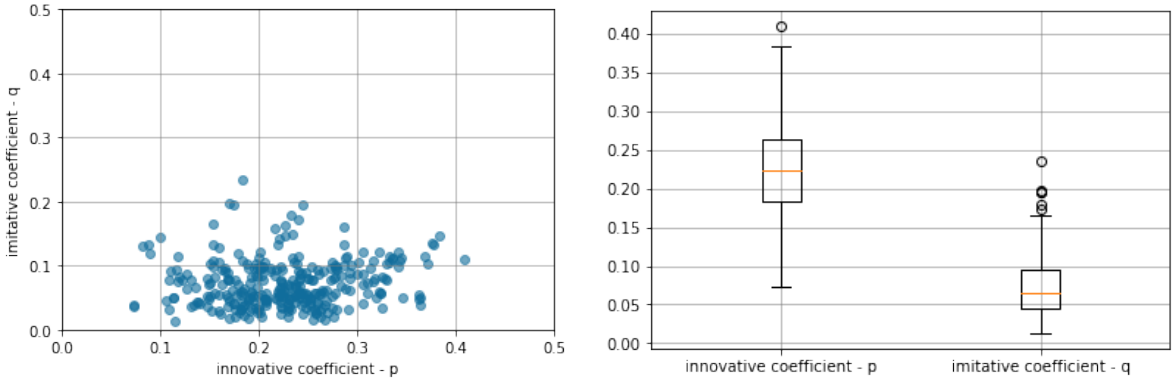

Figure 3 **Distribution of innovative coefficient p and imitative coefficient q. Left: Boxplot of two parameters. Right: Each point in the scatter plot represents a content category.**

6.3. Mean-field approximation and comparison with BDM

In this section, we show that the deterministic approximation of our diffusion model can be a good fit for the diffusion process of online content.

6.3.1. Mean-field approximation As characterizing the diffusion curve for a stochastic model without large-scale simulation is difficult, our model can serve as a deterministic approximation using the mean-field approach. As (1) and (2) show, we can derive the expectations of new innovators and imitators conditioning on the previous adoption history and promotion intensity. However, this is not the expected trajectory in general. When the promoting probability is fixed, the mean-field approximation considers the conditional expectation as a fixed trajectory $\mathcal{H}^{\text{mf}}(t) = \{(a^{\text{mf-in}}(\tau), a^{\text{mf-im}}(\tau))\}_{\tau=0}^{t}$. Therefore, the dynamics of the cumulative number of innovators (resp. imitators) $a^{\text{mf-in}}(t)$ (resp. $a^{\text{mf-im}}(t)$) can be written as

$$a^{\text{mf-in}}(t) = \begin{cases} 0 & t = 0 \\ \mathbb{E}[(a^{\text{in}}(t)|\mathcal{H}^{\text{mf}}(t-1), x(t)] & t > 0 \end{cases},$$

$$a^{\text{mf-im}}(t) = \begin{cases} 0 & t = 0 \\ \mathbb{E}[(a^{\text{im}}(t)|\mathcal{H}^{\text{mf}}(t-1), x(t)] & t > 0 \end{cases}$$

We use synthetic data and simulations to show how the promotion probability and market size impact the diffusion as well as its approximation.

Experiment setup: We consider the diffusion of a specific content. Let the innovative coefficient and imitative coefficient be $p = 0.23$ and $q = 0.07$. For comparison purposes, the market size takes a value in $\{100, 1000, 10000\}$, and the platform maintains the probability of exposing the content to each non-adopter at each time step as a constant that takes a value in $\{0.1, 0.5, 1\}$.

Experiment result: Figure 4 shows that the deterministic approximation can reveal the diffusion process on average. We observe that the simulation and mean field are almost indistinguishable. When the market size is large, the corresponding confidence interval decreases.

6.3.2. Fit true diffusion process with diffusion model We then choose two true diffusion trajectories from the real-world dataset and fit them with ODM and BDM. To fit our model, we use the diffusion parameters estimated using all the exposure and adoption history. To fit the BDM, we estimated the corresponding parameters using nonlinear least squares (NLS). In traditional BDM, the market size cannot be foreseen. Therefore, it is also a parameter that needs to be estimated. We follow the same method as in Srinivasan and Mason (1986) and estimate the market size using the time sequence of adopters.

Experiment setup: For illustration purposes, we consider the setting with two types of users referring to the sub-populations divided by gender. The diffusion parameters used are listed in Table 1.

Experiment result: Figure 5 shows the diffusion curves for both models as well as the true diffusion curves. We observe that the curve of BDM deviates drastically from the true curve. One of the main reasons is that the diffusion behavior does not comply with the assumption of BDM

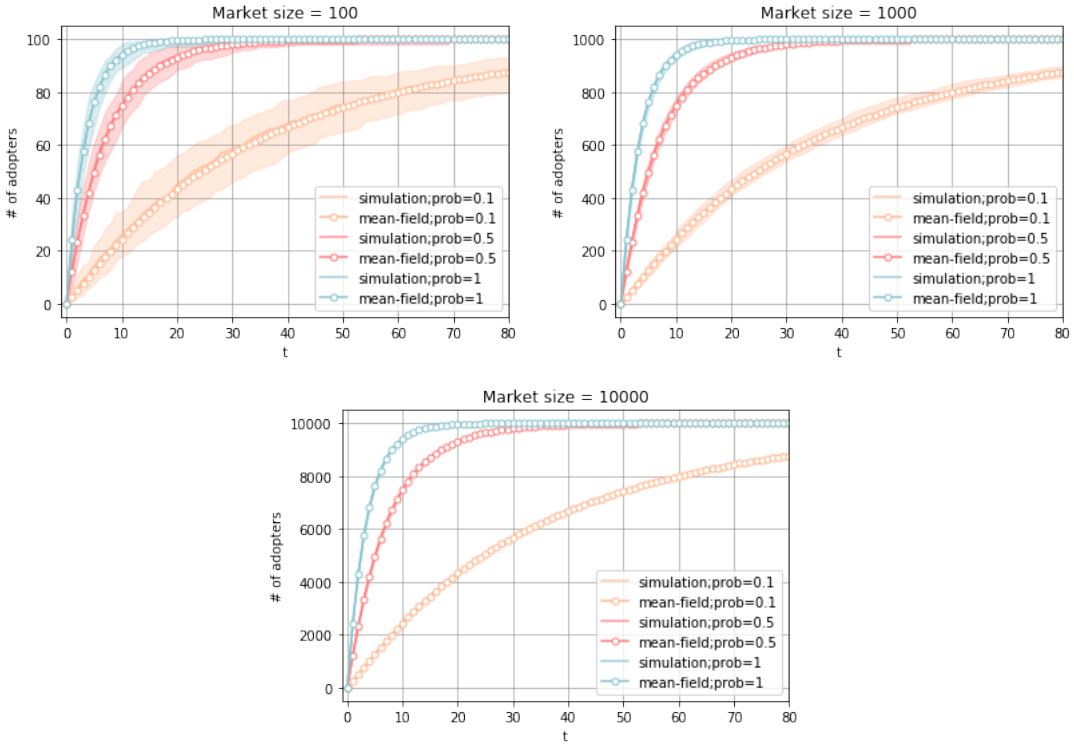

Figure 4 **Mean-field approximation versus simulation (The colored shades representing 95% confidence interval for the number of adopters)**

Table 1 **Diffusion parameters and results for experiment**

Diffusion Coefficients	Content 1		Content 2	
	ODM	BDM	ODM	BDM
Market size m	174978415	19700	174978415	236900
Coefficient p	0.12	1.984e-05	0.24	9.614e-03
Imitative Coefficient q	0.013	0.62	0.044	0.13
MAE	5902.40	22819.04	27560.94	127516.60

thereby causing the estimation of market size to become much smaller than the ground-true value. In order words, the diffusion of online content is complicated by the promotion, and the simple dynamics of BDM are no longer sufficient. In contrast, the ODM curves capture the major trend of diffusion quite well and we observe the MAE of ODM is much smaller than BDM in both cases.

6.4. Experiments on the offline policy

In this section, we use the parameters estimated from the dataset as our ground truth and simulate user behavior as well as the platform's decision with different offline policies. For all experiments in this section, we use the same collection of content category Ω as the real-world dataset and assume the distribution to be uniform with the estimated parameters to be the ground truth.

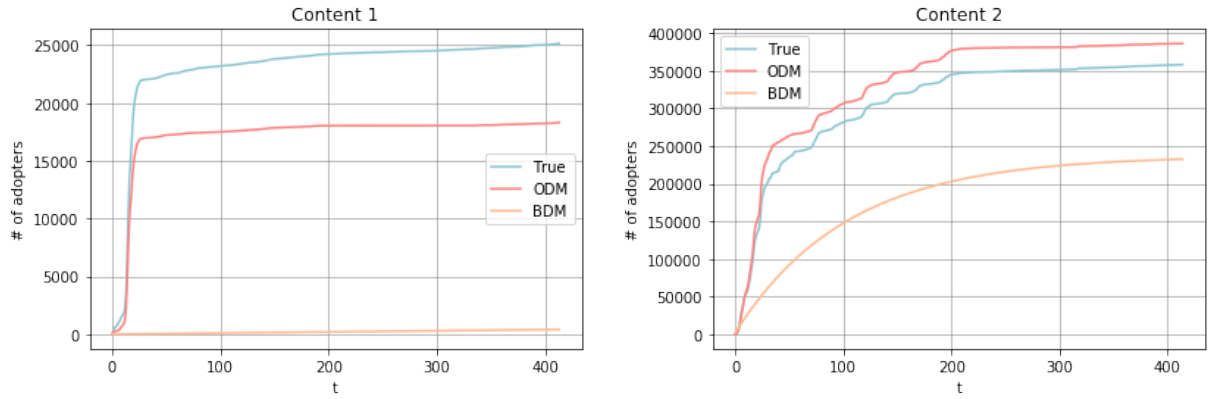

Figure 5 Diffusion curves for two online contents and the corresponding fitted ODM/BDM curves

6.4.1. Robustness and efficiency of DTLU In our numerical experiments, we investigate the robustness and efficiency of our algorithm by comparing how the optimality gap and CPU execution time change with respect to the candidate set size K and the entire content corpus size $|\mathcal{V}|$.

Experiment setup: Set $L = 500$ and $m = 10000$. For the experiments with respect to K, we set $|\mathcal{V}| = 5000$, and K takes values from 550 to 1000 with an increasing interval of 50. For the experiments with respect to K, we fix $K = 1000$, and $|\mathcal{V}|$ takes values from 2000 to 20000 with an increasing interval of 2000. For each test instance, we tested over 50 randomly generated diffusion states $(\boldsymbol{R}, \tilde{\boldsymbol{R}})$.

Tests are performed over the approximation results of the DTLU algorithm with $\epsilon = \{0, 0.5, 0.9\}$ and the optimal results of the mixed integer programming (MIP) in the GUROBI solver.

Experiment result: Figure 6 shows the optimality gap for different test instances. On average, DTLU algorithm has a small optimality gap. When $\epsilon = 0$, the average optimality gap is less than 0.03 in all the cases, implying that the approximation ratio is usually not tight. Figure 7 shows the execution time in CPU seconds. We observe that the runtime of DTLU grows almost linearly with K and $|\mathcal{V}|$. On the contrary, solver takes much longer to solve the MIP problem and the runtime will grow exponentially as the scale of problem increasing. Thus, our algorithm has great advantages in practice. The numerical results are presented in Table 3, and Table 4 in Appendix C .

6.4.2. Compare DTLU performance with benchmarks We evaluate the cumulative reward of the DTLU greedy algorithm over the time periods $\mathcal{T} = \{1, 2, \ldots, 50\}$ with an upper-bound policy and three benchmark policies.

Upper-bound solution: A natural upper-bound solution for the offline CGP problem is to consider an optimization problem similar to (4), except for the cardinality constraint (4b). As

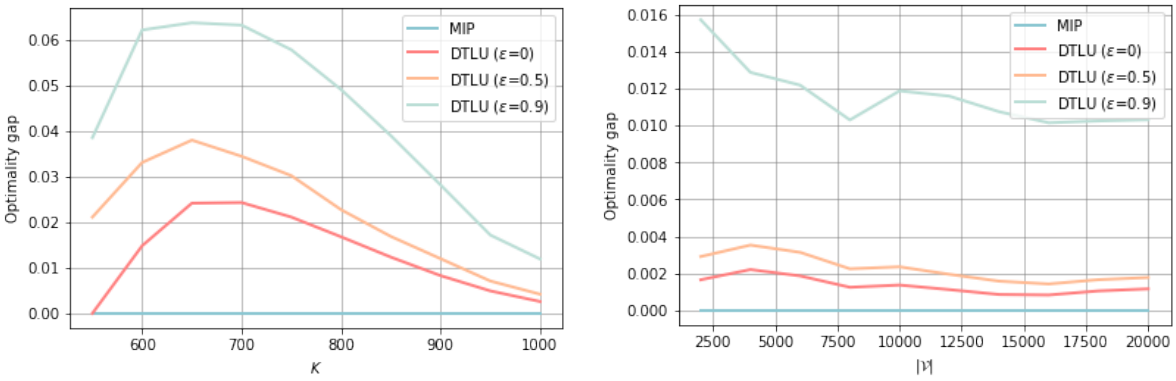

Figure 6 **Robustness test: Left is the result for different K and right is the result for different $|\mathcal{V}|$**

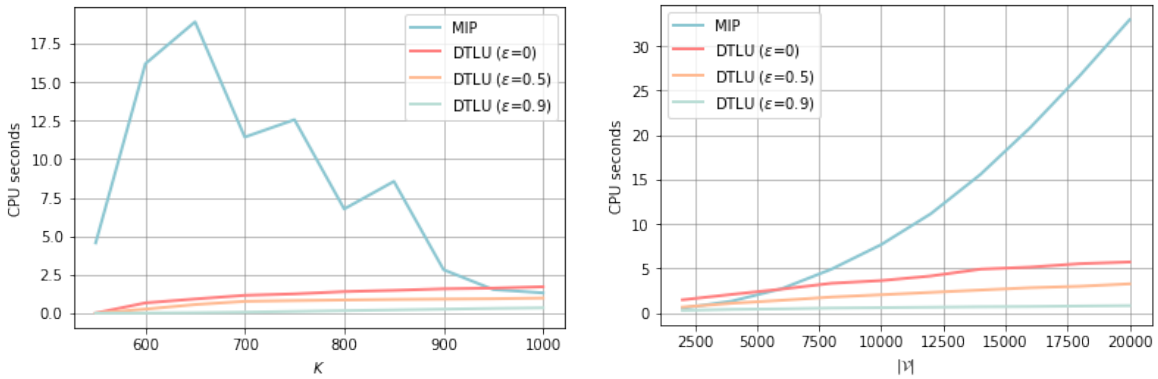

Figure 7 **Efficiency test: Left is the result for different K and right is the result for different $|\mathcal{V}|$**

excluding (4b) also makes the optimization problem no longer NP-hard, the upper-bound solution can be efficiently computed.

Three benchmarks: There are no existing benchmarks for this problem. As our algorithm considers the potential improvement from both the candidate selection and diffusion effect, which are generally overlooked in industry practice, we define two benchmarks that replace either part of the DTLU algorithm with other heuristic methods. The third benchmark replaces both the parts using heuristics.

Benchmark 1 utilizes a heuristic candidate generation strategy: selecting K contents that are newly added into the corpus. It is a common practice in real-world applications to include content that is the newest to users. The promoting probability can then be derived by plugging the candidate set to the parameterized problem. Thus, benchmark 1 can be efficiently computed using Algorithm 1.

Benchmark 2 is a policy that does not consider network effect. As our model acts as a centralized system among all population, most existing decentralized strategies that promote the contents

ignore the interaction between users. Specifically, benchmark 2 considers an optimization problem with the objective

$$\max_{\boldsymbol{x}} \sum_{v \in \mathcal{V}} p_v x_v$$

with the same set of constraints as in (4). As this optimization problem has a property similar to the original problem, we can also apply the DTLU algorithm to solve benchmark 2 with the same approximation ratio.

Benchmark 3 uses the heuristic candidate generation strategy to select K contents that have the least adoption level into the candidate set and then determines the promoting probability by realizing the maximal innovative reward.

Experiment setup: We consider a setting with $K = 1000$, $L = 500$ and $m = 10000$. The contents are considered to be updated at a constant speed, i.e., 300 new contents are added to the corpus every time period.

Experiment result: Figure 8 shows the cumulative average adoption per user for different policies. The numerical results are presented in Table 2. Although our algorithm is non-anticipatory, it can also achieve a good performance along the time horizon. The gap between the upper-bound solution and the DTLU is only approximately 2.30%. Further, DTLU outperforms three benchmarks by 10.92%, 3.57%, and 33.36%, demonstrating considerable improvement. This result highlights the significance of considering both the candidate generation procedure and the network diffusion effect for the platform.

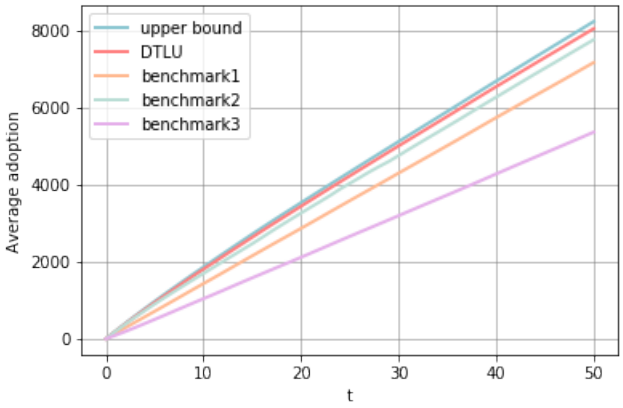

Figure 8 Offline policy performance: compared with upper bound solution and two benchmarks

Table 2 Offline algorithms performance

Offline Algorithm	Average adoption	Relative difference percentage
Upper bound	8244.69	2.30%
DTLU	8059.65	0.00%
Benchmark 1	7179.69	-10.92%
Benchmark 2	7772.30	-3.57%
Benchmark 3	5371.10	-33.36%

6.5. Experiments on the online policy

In this section, we consider the case wherein the diffusion parameters are not known beforehand. We evaluated the performance of Adaptive-DTLU and compared it with a benchmark algorithm.

Benchmark: Explore-then-Exploit approach is widely used as the benchmark for an online algorithm. We also adopt such approachin our experiment. The entire learning process has two phases: the exploration and exploitation phases. As Lai and Robbins (1985) showed, at least $\mathcal{O}(\log T)$ times should be explored for each arm in the MAB problem. Our case is different from the conventional MAB setting; thus, we consider the first $c \log T$ (c is a hyperparameter) to be the exploration phase. During the exploration phase, the platform will choose from the latest uploaded content, and let the sequence be $\{v_1, v_2, \ldots, v_K\}$. The promotion probability x_i of content v_i is equal to $\min\{\frac{1}{L}(1 - \frac{A_{v_i}}{m}), 1 - \sum_{j=1}^{i} x_i\}$. During the exploitation phase, the algorithm uses previously estimated diffusion parameters to determine the candidate set and promotion probability.

Experiment setup: We used the same setting as the offline experiment except that the platform had no information regarding the ground-truth diffusion parameters.

Experiment result: As the approximation ratio is usually not tight in reality as shown in the previous section, we use a pseudo α-CMRegret for illustration. It uses the reward of the offline DTLU algorithm when $\boldsymbol{p}$ and $\boldsymbol{q}$ are known to replace α times the optimal value in (12). As pseudo α-CMRegret is an upper bound of α-CMRegret, it is more comparable in measuring the performance. Figure 9 shows the pseudo α-CMRegret per user for both algorithms. The regret of Adaptive-DTLU is 272.59 while that of the benchmark is 521.12. We notice that the regret has a sharp turning point at the beginning. The reason is that, instead of considering the customer arrives sequentially as the conventional online learning problem, we consider the adoption of the entire market at each time and the estimation shows a significant change in the first few steps. Overall, the regret curve of Adaptive-DTLU grows in a sub-linear shape and the value is only half of the benchmark case.

7. Conclusion

In this study, we consider the content promotion problem under the network diffusion effect for online content platform. Users can adopt the content either from direct targeted promotion by

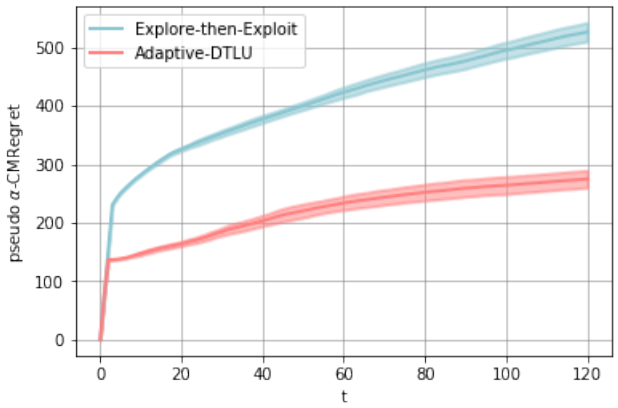

Figure 9 Online policy performance (the colored shades representing 95% confidence interval for the pseudo α-CMRegret adopters

the platform or from sharing by their friends. To maximize the total adoption reward including both types of adoptions, the platform needs to determine what contents to promote with a limited capacity as well as their promotion probability to different types of users.

We propose a novel diffusion model to capture the diffusion behavior of online content platforms and solve the candidate generation and promotion problem under such diffusion model. For the offline optimization problem that is NP-hard, we provide an optimal algorithm for its parametric variant when the candidate set is given and then combined with the submodularity, and propose a $(1-\frac{1}{e})(1-\epsilon)$-approximation algorithm with $\mathcal{O}(\frac{|\mathcal{V}|}{\epsilon}\log(\frac{|\mathcal{V}|}{\epsilon})K)$ time complexity. For the online problem, we present an adaptive algorithm with α-CMRegret upper bounded by $\mathcal{O}(\sqrt{mLT\log T})$. Finally, we used a real-world dataset to validate our model and evaluate the performance of the algorithms. In particular, we found empirical evidences that diffusion parameters vary across different content categories. We also investigated the algorithms using benchmark comparison.

There are several future directions for this study. First, among the population, users may have different preferences for the same content. It is promising to extend the model to multiple user types. However, the nice structure of parametric variants no longer exists, and further exploration is required to efficiently solve the offline problem. Second, online content platforms have abundant user and content information which offers a chance to consider the problem in a contextual setting. Third, when more assumptions are made on the content upload rules, considering the offline problem in multiple periods will be interesting. Ideas such as predict then optimize may help in such case and contribute to an better overall performance.

References

Ahmed, A., Varakantham, P., Lowalekar, M., Adulyasak, Y., and Jaillet, P. (2017). Sampling based approaches for minimizing regret in uncertain markov decision processes (mdps). *Journal of Artificial Intelligence Research*, 59:229–264.

Arora, A., Galhotra, S., and Ranu, S. (2017). Debunking the myths of influence maximization: An in-depth benchmarking study. In *Proceedings of the 2017 ACM international conference on management of data*, pages 651–666.

Azaria, A., Hassidim, A., Kraus, S., Eshkol, A., Weintraub, O., and Netanely, I. (2013). Movie recommender system for profit maximization. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 121–128.

Badanidiyuru, A. and Vondrák, J. (2014). Fast algorithms for maximizing submodular functions. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 1497–1514. SIAM.

Bakshy, E., Hofman, J. M., Mason, W. A., and Watts, D. J. (2011). Everyone's an influencer: quantifying influence on twitter. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 65–74.

Bass, F. M. (1969). A new product growth for model consumer durables. *Management science*, 15(5):215–227.

Chen, W., Wang, Y., and Yuan, Y. (2013). Combinatorial multi-armed bandit: General framework and applications. In *International Conference on Machine Learning*, pages 151–159.

Covington, P., Adams, J., and Sargin, E. (2016). Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM conference on recommender systems*, pages 191–198.

Easingwood, C. J., Mahajan, V., and Muller, E. (1983). A nonuniform influence innovation diffusion model of new product acceptance. *Marketing Science*, 2(3):273–295.

Goel, S., Anderson, A., Hofman, J., and Watts, D. J. (2016). The structural virality of online diffusion. *Management Science*, 62(1):180–196.

Horsky, D. and Simon, L. S. (1983). Advertising and the diffusion of new products. *Marketing Science*, 2(1):1–17.

Kempe, D., Kleinberg, J., and Tardos, É. (2003). Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146.

Kiesling, E., Günther, M., Stummer, C., and Wakolbinger, L. M. (2012). Agent-based simulation of innovation diffusion: a review. *Central European Journal of Operations Research*, 20(2):183–230.

Kitts, B., Freed, D., and Vrieze, M. (2000). Cross-sell: a fast promotion-tunable customer-item recommendation method based on conditionally independent probabilities. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 437–446.

Lai, T. L. and Robbins, H. (1985). Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22.

Leskovec, J., Krause, A., Guestrin, C., Faloutsos, C., VanBriesen, J., and Glance, N. (2007). Cost-effective outbreak detection in networks. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 420–429.

Li, L., Chu, W., Langford, J., and Schapire, R. E. (2010). A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670.

Li, Y., Fan, J., Wang, Y., and Tan, K.-L. (2018). Influence maximization on social graphs: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 30(10):1852–1872.

Lu, W., Chen, S., Li, K., and Lakshmanan, L. V. (2014). Show me the money: dynamic recommendations for revenue maximization. *Proceedings of the VLDB Endowment*, 7(14):1785–1796.

Macal, C. M. (2010). To agent-based simulation from system dynamics. In *Proceedings of the 2010 Winter Simulation Conference*, pages 371–382. IEEE.

Nemhauser, G. L., Wolsey, L. A., and Fisher, M. L. (1978). An analysis of approximations for maximizing submodular set functions—i. *Mathematical programming*, 14(1):265–294.

Norton, J. A. and Bass, F. M. (1987). A diffusion theory model of adoption and substitution for successive generations of high-technology products. *Management science*, 33(9):1069–1086.

Rahmandad, H. and Sterman, J. (2008). Heterogeneity and network structure in the dynamics of diffusion: Comparing agent-based and differential equation models. *Management Science*, 54(5):998–1014.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. (1994). Grouplens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pages 175–186.

Rizoiu, M.-A., Xie, L., Sanner, S., Cebrian, M., Yu, H., and Van Hentenryck, P. (2017). Expecting to be hip: Hawkes intensity processes for social media popularity. In *Proceedings of the 26th International Conference on World Wide Web*, pages 735–744.

Rogers, E. M. (2010). *Diffusion of innovations*. Simon and Schuster.

Shearer, E. and Mitchell, A. (2021). News use across social media platforms in 2020.

Song, L., Tekin, C., and Van Der Schaar, M. (2014). Online learning in large-scale contextual recommender systems. *IEEE Transactions on Services Computing*, 9(3):433–445.

Srinivasan, V. and Mason, C. H. (1986). Nonlinear least squares estimation of new product diffusion models. *Marketing science*, 5(2):169–178.

Sultan, F., Farley, J. U., and Lehmann, D. R. (1990). A meta-analysis of applications of diffusion models. *Journal of marketing research*, 27(1):70–77.

Toubia, O., Goldenberg, J., and Garcia, R. (2008). *A new approach to modeling the adoption of new products: Aggregated diffusion models*. Marketing Science Institute.

Vahabi, H., Koutsopoulos, I., Gullo, F., and Halkidi, M. (2015). Difrec: A social-diffusion-aware recommender system. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 1481–1490.

Zeng, C., Wang, Q., Mokhtari, S., and Li, T. (2016). Online context-aware recommendation with time varying multi-armed bandit. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 2025–2034.

Zhang, H. and Vorobeychik, Y. (2019). Empirically grounded agent-based models of innovation diffusion: a critical review. *Artificial Intelligence Review*, pages 1–35.

Zhao, Q., Erdogdu, M. A., He, H. Y., Rajaraman, A., and Leskovec, J. (2015). Seismic: A self-exciting point process model for predicting tweet popularity. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1513–1522.

Appendix A: Proofs in Section 4

Proof of Theorem 1: Consider a simple case of our problem that no diffusion effect exists which implies $q_v = 0$ for all $v \in \mathcal{V}$. In this case, the objective function $R(\boldsymbol{x})$ is reduced to $R(\boldsymbol{x}) = \sum_{v\in\mathcal{V}} mLp_v x_v$. Let $\overline{s}_v = \frac{1}{L}(1 - \frac{A_v}{m})$, $p_v(m - A_v) = a$ for all $v \in \mathcal{V}$ where a is a constant. Note that, $\overline{s}_v \le 1$ for all $v \in \mathcal{V}$. We claim that

$$R(\boldsymbol{x}^*) \ge ak \iff \text{ there exists } V \subseteq \mathcal{V} \text{ with } |V| = k \text{ and } \sum_{v\in V} \overline{s}_v = 1$$

For the objective value we have

$$R(\boldsymbol{x}) = \sum_{v\in\mathcal{V}} mLp_v x_v \le \sum_{v\in\mathcal{V}} mLp_v \overline{s}_v \cdot \mathbb{1}\{v \in V\} = \sum_{v:v\in V} p_v(m - A_v) \le ak$$

where the first inequality follows from the monotonicity in Lemma 1 and the last inequality follows since V contains at most k values of v. Equality holds if and only if $x_v = \overline{s}_v$ for all $v \in V$. In other words, equality holds if and only if $\sum_{v:v\in V} \overline{s}_v = 1$.

The claim above allows to reduce SUBSET-SUM to our problem as follows. Let positive integers $c_1, \dots c_n$ and C form an instance of SUBSET-SUM. Without loss of generality, we can assume that $c_i \le C$ for all $i \in \{1, 2, \dots, n\}$. Let $\overline{s}_i = c_i/C$. There exists a subset $I \subseteq \{1, \dots, n\}$ such that $\sum_{i\in I} \overline{s}_i = 1$ if and only if the objective value $R(\boldsymbol{x}^*)$ is at least ak for some $k \in \{1, \dots, n\}$. □

Proof of Proposition 1: For a fixed $\mathcal{I}(\boldsymbol{s}, V)$, let $\boldsymbol{x}^*$ and $\boldsymbol{x}'$ be two different optimal solutions. Define $\tilde{\eta}$ and $\tilde{\eta}'$ be the generalized marginal reward that is defined by $\boldsymbol{x}^*$ and $\boldsymbol{x}'$ according to (6), respectively.

If $\boldsymbol{x}^*$ and $\boldsymbol{x}'$ are both optimal, we can find $v_i, v_j \in V$ such that $x^*_{v_i} < x'_{v_i}$ and $x^*_{v_j} > x'_{v_j}$. As a result, the following two inequalities both hold by Lemma 2 but contradict with each other.

$$\tilde{\eta}(\boldsymbol{s}, V) \le \eta^*_{v_j}(\boldsymbol{s}, V) < \eta'_{v_j}(\boldsymbol{s}, V) \le \tilde{\eta}'(\boldsymbol{s}, V)$$
$$\tilde{\eta}'(\boldsymbol{s}, V) \le \eta'_{v_i}(\boldsymbol{s}, V) < \eta^*_{v_i}(\boldsymbol{s}, V) \le \tilde{\eta}(\boldsymbol{s}, V)$$

In conclusion, $\mathcal{I}(\boldsymbol{s}, V)$ can only have a unique optimal solution $\boldsymbol{x}^*(\boldsymbol{s}, V)$. □

Proof of Proposition 2: For a fixed $\mathcal{I}(\boldsymbol{s}, V)$, when $\sum_{v\in V} s_v > 1$, we know the optimal solution satisfies $\sum_{v\in V} x^*_v = 1$. As $\boldsymbol{x}^*$ can be derived as (7), we have

$$\sum_{v\in\overline{V}} x^*_v + \sum_{v\in\tilde{V}} x^*_v + \sum_{v\in\underline{V}} x^*_v = \sum_{v\in\overline{V}} s_v + \sum_{v\in\tilde{V}} \frac{\mu_v - \tilde{\eta}}{2\lambda_v} + \sum_{v\in\underline{V}} 0 = 1$$

However, for all $\eta > \tilde{\eta}$ (resp. $\eta < \tilde{\eta}$), we will have the derived solution $x^\eta_v < x^*_v$ (resp. $x^\eta_v > x^*_v$) holds for all $v \in V$. Thus, $\sum_{v\in V} x^\eta_v$ will not be equal to 1 which contradicts with (8). □

Define $\overline{x}_w(\boldsymbol{s}, V) = x^*_w(\boldsymbol{s} + (\overline{s}_w - s_w)\boldsymbol{e}_w, V)$ which is the optimal solution of w when its parameter attains the upper bound while keeping other parameters the same.

LEMMA 7. *For any instance $\mathcal{I}(\boldsymbol{s}, V)$, the following properties hold for all $w \in V$.*

(i) If $s_w \ge \overline{x}_w(\boldsymbol{s}, V)$, $U(\boldsymbol{s}, V) = U(\boldsymbol{s} + (\overline{s}_w - s_w)\boldsymbol{e}_w, V)$.

(ii) If $s_w < \overline{x}_w(\boldsymbol{s}, V)$, $x^*_w(\boldsymbol{s}, V) = s_w$.

Proof of Lemma 7: We consider two cases $s_w \geq \overline{x}_w$ and $s_w < \overline{x}_w$, respectively.

Proof of (i): It is obvious that $U(\boldsymbol{s}+(\overline{s}_w - s_w)\boldsymbol{e}_w, V) \geq U(\boldsymbol{s}, V)$. When $s_w \geq \overline{x}_w(\boldsymbol{s}, V)$, $\boldsymbol{x}^*(\boldsymbol{s}+(\overline{s}_w - s_w)\boldsymbol{e}_w, V)$ is a feasible solution for $\mathcal{I}(\boldsymbol{s}, V)$ and consequently $U(\boldsymbol{s}+(\overline{s}_w - s_w)\boldsymbol{e}_w, V) \leq U(\boldsymbol{s}, V)$. Thus, property (i) holds.

Proof of (ii): When $\sum_{v\in V} s_v \leq 1$, it is trivial that $x_w^*(\boldsymbol{s}, V) = s_w$.

When $\sum_{v\in V} s_v > 1$, assume $x_w^*(\boldsymbol{s}, V) < s_w$ and we will prove by contradiction. The condition that $\sum_{v\in V} s_v > 1$ implies $\sum_{v\in V} x_v^*(\boldsymbol{s}, V) = 1$. Let $\boldsymbol{x}_\alpha = \alpha \boldsymbol{x}^*(\boldsymbol{s}, V) + (1-\alpha)\boldsymbol{x}^*(\boldsymbol{s}+(\overline{s}_w - s_w)\boldsymbol{e}_w, V)$ for $\alpha \in (0,1)$. When α is sufficiently close to 1, we can have $\boldsymbol{x}_\alpha$ satisfies all the constraints for $\mathcal{I}(\boldsymbol{s}, V)$. Since the objective function $R(\boldsymbol{x})$ is concave with regard to $\boldsymbol{x}$, $\alpha f(\boldsymbol{x}^*(\boldsymbol{s}, V)) + (1-\alpha) f(\boldsymbol{x}^*(\boldsymbol{s}+(\overline{s}_w - s_w)\boldsymbol{e}_w) \leq f(\boldsymbol{x}_\alpha)$. It contradicts with $f(\boldsymbol{x}_\alpha) < f(\boldsymbol{x}^*(\boldsymbol{s}, V)) \leq f(\boldsymbol{x}^*(\boldsymbol{s}+(\overline{s}_w - s_w)\boldsymbol{e}_w, V)$. Thus, property (ii) holds. □

Define $\Delta U(\boldsymbol{s}, \underline{c}, \overline{c}, V, w) = U(\boldsymbol{s}+(\overline{c}-s_w)\boldsymbol{e}_w, V\cup\{w\}) - U(\boldsymbol{s}+(\underline{c}-s_w)\boldsymbol{e}_w, V\cup\{w\})$ which denotes the difference between the optimal values of two instances where the parameters of w are $\underline{c}$ and $\overline{c}$, respectively.

LEMMA 8. *For all $\boldsymbol{s}\in\mathcal{S}$, $V_1 \subseteq V_2 \subseteq \mathcal{V}$ and $w \in \mathcal{V}\backslash V_2$, the following inequality holds.*

$$\Delta U(\boldsymbol{s}, 0, s_w, V_2, w) \leq \Delta U(\boldsymbol{s}, 0, s_w, V_1, w) \tag{14}$$

Proof of Lemma 8: Let $a_1 = 1 - \sum_{v\in V_1} s_v$ and $a_2 = \overline{x}_w(\boldsymbol{s}, V_2 + \{w\})$. We first decompose LHS and RHS of (14) into three parts by the value of parameter for content w at a_1 and a_2, and then show the inequality holds for each part separately. Without loss of generality, we consider the case when $0 < a_1 < a_2 < s_w$. Other cases can be easily generalized. Specifically, we have

$$\text{LHS} = \Delta U(\boldsymbol{s}, 0, a_1, V_2, w) + \Delta U(\boldsymbol{s}, a_1, a_2, V_2, w) + \Delta U(\boldsymbol{s}, a_2, s_w, V_2, w)$$
$$\text{RHS} = \Delta U(\boldsymbol{s}, 0, a_1, V_1, w) + \Delta U(\boldsymbol{s}, a_1, a_2, V_1, w) + \Delta U(\boldsymbol{s}, a_2, s_w, V_1, w)$$

We will show the following claims:

(i) $\Delta U(\boldsymbol{s}, a_2, s_w, V_2, w) \leq \Delta U(\boldsymbol{s}, a_2, s_w, V_1, w)$.

(ii) $\Delta U(\boldsymbol{s}, 0, a_1, V_2, w) \leq \Delta U(\boldsymbol{s}, 0, a_1, V_1, w)$.

(iii) $\Delta U(\boldsymbol{s}, a_1, a_2, V_2, w) \leq \Delta U(\boldsymbol{s}, a_1, a_2, V_1, w)$.

Proof of (i): By Lemma 7 (i), we directly have $\Delta U(\boldsymbol{s}, a_2, s_w, V_2, w) = 0 \leq \Delta U(\boldsymbol{s}, a_2, s_w, V_1, w)$ holds.

By Lemma 7 (ii), we have $x_w^*(\boldsymbol{s}+(a-s_w)\boldsymbol{e}_w, V_2+\{w\}) = a$ for all $0 \leq a < a_2$. Further, $x_w^*(\boldsymbol{s}+(a-s_w)\boldsymbol{e}_w, V_1+\{w\}) \geq x_w^*(\boldsymbol{s}+(a-s_w)\boldsymbol{e}_w, V_2+\{w\})$ holds by Theorem 2. Thus, $x_w^*(\boldsymbol{s}+(a-s_w)\boldsymbol{e}_w, V_2+\{w\}) = x_w^*(\boldsymbol{s}+(a-s_w)\boldsymbol{e}_w, V_2+\{w\}) = a$ for the following two cases.

Proof of (ii): On one hand, we can notice that $x_v^*(\boldsymbol{s}+(a-s_w)\boldsymbol{e}_w, V_2+\{w\}) = s_v$ for all $v \in V_1$ when $0 \leq a < a_1$, which means $\Delta U(\boldsymbol{s}, 0, a_1, V_1, w) = -\lambda_w a_1^2 + \mu_w a_1$. On the other hand, we can notice that $\Delta U(\boldsymbol{s}, 0, a_1, V_2, w) \leq -\lambda_w a_1^2 + \mu_w a_1$ as $1 - \sum_{v\in V_2} s_v \leq a_1$. Thus, $\Delta U(\boldsymbol{s}, 0, a_1, V_2, w) \leq \Delta U(\boldsymbol{s}, 0, a_1, V_1, w)$ holds.

Proof of (iii): For any $a_1 < a < a_2$, we first consider the instance $\mathcal{I}(\boldsymbol{s}+(a-s_w)\boldsymbol{e}_w, V+\{w\})$ and the parameters are discard for notation simplicity. By (8), we can derive the value of $\tilde{\eta}$ as

$$\tilde{\eta} = \frac{\sum_{v\in\tilde{V}} \frac{\mu_v}{2\lambda_v} - 1 + \sum_{v\in\overline{V}} s_v + a}{\sum_{v\in\tilde{V}} \frac{1}{2\lambda_v}}$$

The total reward gained by the contents in $\tilde{V}$ can be represented by

$$\sum_{v\in\tilde{V}} -\lambda_v x^{*2} + \mu_v x^* = \sum_{v\in\tilde{V}} -\lambda_v \left(\frac{\mu_v - \tilde{\eta}}{2\lambda_v}\right)^2 + \mu_v \frac{\mu_v - \tilde{\eta}}{2\lambda_v} = \sum_{v\in\tilde{V}} \frac{\mu_v^2 - \tilde{\eta}^2}{4\lambda_v}$$

Meanwhile, we also consider the instance with a small perturbation on the parameter of content w, that is, $\mathcal{I}(\boldsymbol{s} + (a - s_w + \epsilon)\boldsymbol{e}_w, V + \{w\})$, where ϵ is a positive number. To distinguish from the previous instance, we use the subscript ϵ to denote all the variables in the perturbed instance. When ϵ is small enough, $\overline{V}_\epsilon$ and $\tilde{V}_\epsilon$ keep the same as $\overline{V}$ and $\tilde{V}$. The value of $\tilde{\eta}_\epsilon$ can be derived as

$$\tilde{\eta}_\epsilon = \frac{\sum_{v\in\tilde{V}} \frac{\mu_v}{2\lambda_v} - 1 + \sum_{v\in\overline{V}} s_v + a + \epsilon}{\sum_{v\in\tilde{V}} \frac{1}{2\lambda_v}} = \tilde{\eta} + \frac{1}{\sum_{v\in\tilde{V}} \frac{1}{2\lambda_v}}\epsilon$$

Therefore, the difference of optimal reward value for $\mathcal{I}(\boldsymbol{s} + (a - s_w + \epsilon)\boldsymbol{e}_w, V + \{w\})$ and $\mathcal{I}(\boldsymbol{s} + (a - s_w)\boldsymbol{e}_w, V + \{w\})$ can be calculated as

$$\begin{aligned}
\Delta U(\boldsymbol{s}, a, a+\epsilon, V, w) &= \sum_{v\in\tilde{V}_\epsilon} \frac{\mu_v - \tilde{\eta}_\epsilon^2}{4\lambda_v} - \sum_{v\in\tilde{V}} \frac{\mu_v - \tilde{\eta}^2}{4\lambda_v} \\
&= \left(\sum_{v\in\tilde{V}} \frac{1}{4\lambda_v}\right)\left(-2\tilde{\eta}\frac{1}{\sum_{v\in\tilde{V}} \frac{1}{2\lambda_v}}\epsilon + \frac{1}{(\sum_{v\in\tilde{V}} \frac{1}{2\lambda_v})^2}\epsilon^2\right) \\
&= -\tilde{\eta}\epsilon + \frac{1}{\sum_{v\in\tilde{V}} \frac{1}{\lambda_v}}\epsilon^2
\end{aligned}$$

Note that, the difference of reward only comes from the contents in $\tilde{V}$, since contents in $\overline{V}$ and $\underline{V}$ keeps optimal solutions to be the same in both cases.

We can then measure the difference between $\Delta U(\boldsymbol{s}, a, a+\epsilon, V_1, w)$ and $\Delta U(\boldsymbol{s}, a, a+\epsilon, V_2, w)$ with regard to ϵ. By taking the limit of ϵ, we have

$$\lim_{\epsilon\to 0} \frac{\Delta U(\boldsymbol{s}, a, a+\epsilon, V_2, w) - \Delta U(\boldsymbol{s}, a, a+\epsilon, V_1, w)}{\epsilon} = \tilde{\eta}(\boldsymbol{s} + (a - s_w)\boldsymbol{e}_w, V_1 + \{w\}) - \tilde{\eta}(\boldsymbol{s} + (a - s_w)\boldsymbol{e}_w, V_2 + \{w\}) \le 0$$

The inequality follows from Corollary 1. Therefore, by integral a over $[a_1, a_2]$, $\Delta U(\boldsymbol{s}, a_1, a_2, V_2, w) \le \Delta U(\boldsymbol{s}, a_1, a_2, V_1, w)$ holds.

In conclusion, $\Delta U(\boldsymbol{s}, 0, s_w, V_2, w) \le \Delta U(\boldsymbol{s}, 0, s_w, V_1, w)$. □

Proof of Theorem 3: It is trivial that $U(\boldsymbol{s}, V_1) \le U(\boldsymbol{s}, V_2)$ when $V_1 \subseteq V_2$, thus, $U(\boldsymbol{s}, V)$ is monotone.

To show the submodularity, it suffices to show that

$$U(\boldsymbol{s}, V_2 + \{w\}) - U(\boldsymbol{s}, V_2) \le U(\boldsymbol{s}, V_1 + \{w\}) - U(\boldsymbol{s}, V_1)$$

where $V_1 \subseteq V_2$ and $w \in \mathcal{V}\backslash V_2$. Since adding a content w to the selected subset is equivalent to lifting s_w from 0 to s_w, we just need to show that the increasing reward of lifting one entry of parameter $\boldsymbol{s}$ is diminishing for nested subsets. By Lemma 8, we have

$$U(\boldsymbol{s}, V_2 + \{w\}) - U(\boldsymbol{s} - s_w\boldsymbol{e}_w, V_2) \le U(\boldsymbol{s}, V_1 + \{w\}) - U(\boldsymbol{s} - s_w\boldsymbol{e}_w, V_1)$$

In conclusion, $U(\boldsymbol{s}, V)$ is monotone submodular with regard to V. □

Appendix B: Proofs in Section 5

Proof of Lemma 3: Innovative adoptions of contents belonging to category ω by time t can be considered as a Binomial random variable with parameters $N_\omega(t)$ and p_ω. Thus, it's obvious that $\hat{p}_\omega(t)$ is an unbiased estimator for $p_\omega(t)$.

Imitative adoptions towards contents belonging to category ω by time t are is more complicated. When all the diffusion states are given, it can be considered as the summation of a sequence of independent Bernoulli variables that are not identically distributed. Specifically, the sequence consists of $M_\omega(\tau)$ Bernoulli random variables with success probability $q_\omega \frac{\tilde{M}_\omega(\tau)}{m M_\omega(\tau)}$ for all $\tau \in [1,t]$ and v that belongs to category ω. Therefore, the total imitative adoption by t is a Poisson binomial variable whose expected value is $q_\omega \frac{\sum_{\tau=1}^t \tilde{M}_\omega(t)}{m}$. Thus, $\hat{q}_\omega(t)$ is an unbiased estimator for q_ω. □

Proof of Lemma 4: Using Hoeffding's inequality, we have

$$\begin{aligned}
&\mathbb{P}\left(p_\omega^{\mathrm{OP}}(t) < p_\omega\right) + \mathbb{P}\left(p_\omega^{\mathrm{OP}}(t) > p_\omega + 2\sqrt{\frac{2\log t}{N_\omega(t)}}\right) \\
&= \mathbb{P}\left(\hat{p}_\omega(t) < p_\omega - \sqrt{\frac{2\log t}{N_\omega(t)}}\right) + \mathbb{P}\left(\hat{p}_\omega(t) > p_\omega + \sqrt{\frac{2\log t}{N_\omega(t)}}\right) \\
&= \mathbb{P}\left(|\hat{p}_\omega(t) - p_\omega| > \sqrt{\frac{2\log t}{N_\omega(t)}}\right) \le 2e^{-4\log t} = \frac{2}{t^4}
\end{aligned}$$

Similarly, we can get the concentration bound of $\hat{q}_\omega(t)$,

$$\begin{aligned}
\mathbb{P}\left(|\hat{q}_\omega(t) - q_\omega| \ge \frac{m}{\sum_{\tau=1}^t \tilde{M}_\omega(\tau)}\sqrt{\frac{2\log t}{\sum_{\tau=1}^t M_\omega(\tau)}}\right) &= \mathbb{P}\left(|\hat{q}_\omega(t) - q_\omega| \frac{\sum_{\tau=1}^t \tilde{M}_\omega(\tau)}{m} \ge \sqrt{\frac{2\log t}{\sum_{\tau=1}^t M_\omega(\tau)}}\right) \\
&\le 2e^{-4\log t} = \frac{2}{t^4}
\end{aligned}$$

□

Proof of Lemma 5:

$$\begin{aligned}
&R_{\boldsymbol{p}',\boldsymbol{q}'}(\boldsymbol{x};\boldsymbol{A},\tilde{\boldsymbol{A}}) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x};\boldsymbol{A},\tilde{\boldsymbol{A}}) \\
&= \sum_{\omega\in\Omega}\sum_{v\in S_\omega} [mLx_v^2(-q'_\omega p'^2_\omega + q_\omega p_\omega^2) + Lmx_v(p'_\omega - p_\omega) + Lx_v(m-1-(A_v+\tilde{A}_v))(p'_\omega q'_\omega - p_\omega q_\omega) \\
&\quad + Lx_v(p'^2_\omega q'_\omega - p_\omega^2 q_\omega)] \\
&\le \sum_{\omega\in\Omega}\sum_{v\in S_\omega} mLx_v[(p'_\omega - p_\omega) + (1-\frac{1}{m})(p'_\omega q'_\omega - p_\omega q_\omega) + \frac{1}{m}(p'^2_\omega q'_\omega - p_\omega^2 q_\omega)] \qquad (15a) \\
&\le \sum_{\omega\in\Omega}\sum_{v\in S_\omega} mLx_v[\Delta p_\omega + (1-\frac{1}{m})(2\Delta p_\omega + \Delta q_\omega) + \frac{1}{m}(6\Delta p_\omega + \Delta q_\omega)] \qquad (15b) \\
&= \sum_{\omega\in\Omega}\sum_{v\in S_\omega} mLx_v[(3+\frac{4}{m})\Delta p_\omega + \Delta q_\omega]
\end{aligned}$$

where (15a) follows since we assume $\boldsymbol{p}' \ge \boldsymbol{p}$ and $\boldsymbol{q}' \ge \boldsymbol{q}$ and (15b) follows since $\boldsymbol{0} \le \Delta\boldsymbol{p}, \Delta\boldsymbol{q} \le \boldsymbol{1}$.

In conclusion, we can have the linear function to be $f(\Delta\boldsymbol{p}, \Delta\boldsymbol{q}; \boldsymbol{x}) = \sum_{\omega\in\Omega}\sum_{v\in S_\omega} mLx_v[(3+\frac{4}{m})\Delta p_\omega + \Delta q_\omega]$. □

Proof of Lemma 6: By assumption 2, we have $\tilde{M}_\omega(1) \geq 2(m-2) \geq 1$. Thus, $\frac{m}{\sum_{\tau=1}^{t}\tilde{M}_\omega(\tau)} \leq 1$ holds for all $t \geq 1$. Furthermore, by constraints (4d), $N_\omega(t) - N_\omega(t-1) \leq M_\omega(t-1)$. Sum up along t, we have $N_\omega(t) - N_\omega(1) \leq \sum_{\tau=1}^{t-1} M_\omega(\tau)$.

In conclusion,

$$\sqrt{\frac{1}{N_\omega(t)-N_\omega(1)}} \geq \frac{m}{\sum_{\tau=1}^{t-1}\tilde{M}_\omega(\tau)}\sqrt{\frac{1}{\sum_{\tau=1}^{t-1}M_\omega(\tau)}}$$

□

Proof of Theorem 4: Define the "large" probability event of each parameter as $E_\omega(t) = \{p_\omega^{\mathrm{OP}}(t) - 2\sqrt{\frac{2\log t}{N_\omega(t)}} \leq p_\omega(t) \leq p_\omega^{\mathrm{OP}}(t)\}$ and $F_\omega(t) = \{q_\omega^{\mathrm{OP}}(t) - 2m\frac{\sqrt{2\log t\sum_{\tau=1}^{t}M_\omega(\tau)}}{\sum_{\tau=1}^{t}\tilde{M}_\omega(\tau)} \leq q_\omega(t) \leq q_\omega^{\mathrm{OP}}(t)\}$. Let $\zeta(t) = \bigcap_{\omega\in\Omega}E_\omega(t)\bigcap_{\omega\in\Omega}F_\omega(t)$ denoting the clean event when the "large" probability events hold simultaneously at time t.

Thus, we have

$$\begin{aligned}\alpha\text{-CMRegret}_{\boldsymbol{p},\boldsymbol{q}}^{\pi}(T) =& \mathbb{E}_\pi\left[\sum_{t=1}^{T} R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}^*;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}_t;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t)\middle|\bigcap_{t=1}^{T}\zeta(t)\right]\mathbb{P}\left(\bigcap_{t=1}^{T}\zeta(t)\right)\\ &+\mathbb{E}_\pi\left[\sum_{t=1}^{T} R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}^*;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}_t;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t)\middle|(\bigcap_{t=1}^{T}\zeta(t))^{\complement}\right]\mathbb{P}\left((\bigcap_{t=1}^{T}\zeta(t))^{\complement}\right)\end{aligned}$$

Let's first consider the clean event. Using the optimality guarantee of the offline policy, the single period regret can be upper bounded by

$$\begin{aligned}&\alpha R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}^*;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}_t;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t)\\ =&\alpha R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}^*;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p}^{\mathrm{OP}}(t),\boldsymbol{q}^{\mathrm{OP}}(t)}(\boldsymbol{x}_t;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t) + R_{\boldsymbol{p}^{\mathrm{OP}}(t),\boldsymbol{q}^{\mathrm{OP}}(t)}(\boldsymbol{x}_t;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}_t;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t)\\ \leq&\alpha R_{\boldsymbol{p}^{\mathrm{OP}}(t),\boldsymbol{q}^{\mathrm{OP}}(t)}(\boldsymbol{x}^*;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t) - \alpha R_{\boldsymbol{p}^{\mathrm{OP}}(t),\boldsymbol{q}^{\mathrm{OP}}(t)}(\boldsymbol{x}_t^*;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t) + R_{\boldsymbol{p}^{\mathrm{OP}}(t),\boldsymbol{q}^{\mathrm{OP}}(t)}(\boldsymbol{x}_t;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}_t;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t)\\ \leq&R_{\boldsymbol{p}^{\mathrm{OP}}(t),\boldsymbol{q}^{\mathrm{OP}}(t)}(\boldsymbol{x}_t;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}_t;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t)\end{aligned}$$

Let $\Phi_u(t)$ be the set of contents that are promoted to user u at time t. Consequently, define $\phi_\omega(t)$ as the total number of times that contents of category ω is promoted to users at time t.

$$\begin{aligned}&\mathbb{E}_\pi\left[\sum_{t=1}^{T} R_{\boldsymbol{p},\boldsymbol{q}}(\tilde{\boldsymbol{x}}_t^*;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}_t;\boldsymbol{A}_t,\tilde{\boldsymbol{A}}_t)\middle|\bigcap_{t=1}^{T}\zeta(t)\right]\\ &\leq \mathbb{E}_\pi\left[\sum_{t=1}^{T} f(\boldsymbol{p}^{\mathrm{OP}}(t)-\boldsymbol{p},\boldsymbol{q}^{\mathrm{OP}}(t)-\boldsymbol{q};\boldsymbol{x}_t)\middle|\bigcap_{t=1}^{T}\zeta(t)\right] &\text{(16a)}\\ &= \mathbb{E}_\pi\left[\sum_{t=1}^{T}\sum_{\omega\in\Omega}\sum_{v\in S_\omega(t)}\sum_{u=1}^{m}\mathbb{1}\{v\in\Phi_u(t)\}[(3+\frac{4}{m})(p_\omega^{\mathrm{OP}}-p_\omega)+(q_\omega^{\mathrm{OP}}-q_\omega)]\middle|\bigcap_{t=1}^{T}\zeta(t)\right]\\ &\leq \mathbb{E}_\pi[\sum_{t=1}^{T-1}\sum_{\omega\in\Omega}\sum_{v\in S_\omega(t)}\sum_{u=1}^{m}\mathbb{1}\{v\in\Phi_u(t)\}[(3+\frac{4}{m})\sqrt{\frac{8\log t}{N_\omega(t)}}+\frac{m}{\sum_{\tau=1}^{t}\tilde{M}_\omega(\tau)}\sqrt{\frac{8\log t}{\sum_{\tau=1}^{t}M_\omega(\tau)}}]+C_1] &\text{(16b)}\\ &\leq C_2\sqrt{\log T}\mathbb{E}_\pi[\sum_{\omega\in\Omega}\sum_{v\in S_\omega(t)}\sum_{u=1}^{m}\sum_{t=1}^{T-1}\mathbb{1}\{v\in\Phi_u(t)\}(\sqrt{\frac{1}{N_\omega(t)}}+\sqrt{\frac{1}{N_\omega(t+1)-N_\omega(1)}})] &\text{(16c)}\\ &= C_2\sqrt{\log T}\mathbb{E}_\pi[\sum_{\omega\in\Omega}\sum_{t=1}^{T-1}\phi_\omega(t)(\sqrt{\frac{1}{N_\omega(t)}}+\sqrt{\frac{1}{N_\omega(t+1)-N_\omega(1)}})]\end{aligned}$$

where C_1, C_2 are two constants. (16a) and (16b) follow from Lemma 5, (16c) follows from Lemma 6. We notice that $\phi_\omega(t) \le Lm$ since each category of content can only be recommended to the users L times during each time period. Also, for all $\forall \omega \in \Omega$, $N_\omega(t) = N_\omega(t-1) + \phi_\omega(t)$. It indicates that

$$\mathbb{E}_\pi\left[\sum_{t=1}^{T}\phi_\omega(t)\sqrt{\frac{1}{N_\omega(t)}}\right] \le \mathbb{E}_\pi\left[mL + \sum_{\ell=1}^{\lceil\frac{N_\omega(T)}{mL}\rceil} mL\sqrt{\frac{1}{mL\ell}}\right] \le C_3\sqrt{N_\omega(T)}$$

$$\mathbb{E}_\pi\left[\sum_{t=1}^{T-1}\phi_\omega(t)\sqrt{\frac{1}{N_\omega(t+1)-N_\omega(1)}}\right] \le \mathbb{E}_\pi\left[mL + \sum_{\ell=1}^{\lceil\frac{N_\omega(T)}{mL}\rceil} mL\sqrt{\frac{1}{mL\ell}}\right] \le C_3\sqrt{N_\omega(T)}$$

Since $\sum_{\omega\in\Omega} N_\omega(T) \le LmT$,

$$\begin{aligned}\mathbb{E}_\pi\left[\sum_{t=1}^{T} R_{\boldsymbol{p},\boldsymbol{q}}(\tilde{\boldsymbol{x}}_t^*; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}^{(t)}; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t)\middle|\bigcap_{t=1}^{T}\zeta(t)\right] &\le C_2\sqrt{\log T}\mathbb{E}_\pi\left[\sum_{\omega\in\Omega} 2C_3\sqrt{N_\omega(T)}\right]\\ &\le C_4\sqrt{mLT\log T}\end{aligned}$$

Consider the "bad" event. Using the union bound to calculate the probability of "bad" event, we have

$$\mathbb{P}\left((\bigcap_{t=1}^{T}\zeta(t))^{\complement}\right) \le \sum_{t=1}^{T}\sum_{\omega\in\Omega}\mathbb{P}\left(E_\omega(t)^{\complement}\right) + \mathbb{P}\left(F_\omega(t)^{\complement}\right) \le C_5\frac{|\Omega|}{T^3}$$

Combine with Assumption 1, we have

$$\mathbb{E}_\pi\left[\sum_{t=1}^{T} R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}_t^*; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t) - R_{\boldsymbol{p},\boldsymbol{q}}(\boldsymbol{x}_t; \boldsymbol{A}_t, \tilde{\boldsymbol{A}}_t)\middle|(\bigcap_{t=1}^{T}\zeta(t))^{\complement}\right]\mathbb{P}\left((\bigcap_{t=1}^{T}\zeta(t))^{\complement}\right) \le C_6 mT^2\frac{|\Omega|}{T^3} = \mathcal{O}(\frac{m|\Omega|}{T})$$

In conclusion,

$$\alpha\text{-CMRegret} = \mathcal{O}(\sqrt{mLT\log T})$$

□

Appendix C: Numerical experiments

Table 3 **Numerical results for different candidate set size K**

K	Optimal		DTLU ($\epsilon=0$)		DTLU ($\epsilon=0.5$)		DTLU ($\epsilon=0.9$)	
	obj	CPU sec.	gap	CPU sec.	gap	CPU sec.	gap	CPU sec.
550	147.00	4.58	1.16e-6	0.02	0.0211	0.01	0.0385	0.01
600	156.72	16.20	0.0148	0.68	0.0331	0.27	0.0621	0.02
650	161.15	18.92	0.0241	0.93	0.0380	0.57	0.0637	0.04
700	163.37	11.44	0.0243	1.17	0.0344	0.78	0.0631	0.08
750	164.54	12.57	0.0211	1.26	0.0302	0.83	0.0577	0.13
800	165.16	6.77	0.0168	1.41	0.0227	0.87	0.0490	0.18
850	165.46	8.57	0.0123	1.49	0.0169	0.90	0.0389	0.22
900	165.57	2.83	0.0083	1.59	0.0120	0.93	0.0281	0.27
950	165.59	1.55	0.0049	1.64	0.0071	0.95	0.0172	0.32
1000	165.60	1.33	0.0026	1.73	0.0042	0.99	0.0119	0.36

Table 4 Numerical results for different content size $|\mathcal{V}|$

$\|\mathcal{V}\|$	Optimal		DTLU ($\epsilon = 0$)		DTLU ($\epsilon = 0.5$)		DTLU ($\epsilon = 0.9$)	
	obj	CPU sec.	gap	CPU sec.	gap	CPU sec.	gap	CPU sec.
2000	153.87	0.54	0.0017	1.48	0.0029	0.66	0.0157	0.29
4000	174.66	1.34	0.0022	2.09	0.0035	1.10	0.0129	0.40
6000	184.09	2.73	0.0019	2.68	0.0031	1.43	0.0122	0.46
8000	189.07	4.93	0.0013	3.34	0.0022	1.79	0.0103	0.56
10000	193.85	7.69	0.0014	3.64	0.0024	2.05	0.0119	0.60
12000	196.93	11.16	0.0011	4.16	0.0020	2.32	0.0116	0.65
14000	199.20	15.59	0.0009	4.93	0.0016	2.58	0.0107	0.70
16000	201.04	20.84	0.0008	5.15	0.0014	2.84	0.0101	0.73
18000	202.71	26.70	0.0011	5.54	0.0017	3.00	0.0102	0.78
20000	204.62	32.97	0.0012	5.73	0.0018	3.28	0.0103	0.82